■ Così costruisci un NAS su misura per te ■ Crea un ambiente di sviluppo virtuale in PHP

LINUX PRO

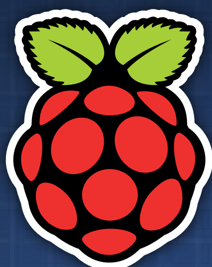
FEBBRAIO 2015

Diventa protagonista della programmazione Open Source

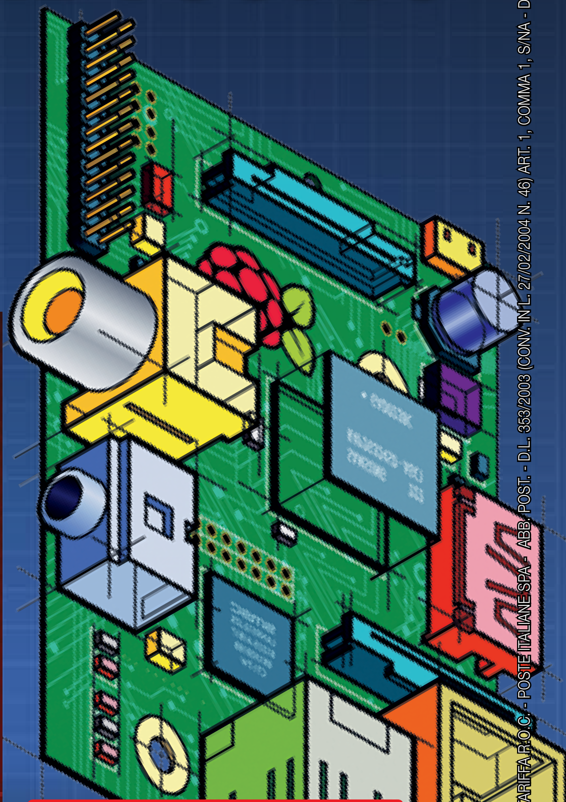


Integrare Docker
Ecco come sfruttare il sistema di contenitori in ambiente di produzione

RASPBERRY PI
FAGLI FARE QUELLO CHE VUOI TU



- CONTROLLO VOCALE
- JUKEBOX IN STREAMING
- OVERCLOCK FACILISSIMO
- E MOLTO ALTRO ANCORA!





TANGO STUDIO

LA DISTRO FATTA PER I PROFESSIONISTI DELLA MUSICA



GHOSTBSD 4.0 - STABILE E FACILE DA USARE

Niente fronzoli!
Basta coi desktop pesanti, ecco quelli fatti per alte prestazioni p.54

L'audio è meglio con JACK
Il sistema di gestione audio che risolve ogni problema p.22

TARIFFA R.O.C. - POSTE ITALIANE SPA - ABB. POST. - D.L. 353/2003 (CONV. IN L. 27/02/2004 N. 46) ART. 1, COMMA 1, S/NA - DISTRIBUTORE: PRESS-DI, MILANO

Linux Pro 150 - 2015 - Mensile - 5,90€



ACCADEMIA DEL CODICE



8 pagine di pura programmazione
da pagina 84



it's a marimo project!

È QUANDO TI SENTI PICCOLO CHE SAI DI ESSERE DIVENTATO GRANDE.

A volte gli uomini riescono a creare qualcosa più grande di loro. Qualcosa che prima non c'era. È questo che noi intendiamo per innovazione ed è in questo che noi crediamo.
Una visione che ci ha fatto investire nel cambiamento tecnologico sempre e solo con l'obiettivo di migliorare il valore di ogni nostra singola produzione.
È questo pensiero che ci ha fatto acquistare per primi in Italia impianti come la rotativa Heidelberg M600 B24. O che oggi, per primi in Europa, ci ha fatto introdurre 2 rotative da 32 pagine Roto-Offset Komori, 64 pagine-versione duplex, così da poter soddisfare ancora più puntualmente ogni necessità di stampa di bassa, media e alta tiratura.
Se crediamo nell'importanza dell'innovazione, infatti, è perché pensiamo che non ci siano piccole cose di poca importanza.
L'etichetta di una lattina di pomodori pelati, quella di un cibo per gatti o quella di un'acqua minerale, un catalogo o un quotidiano, un magazine o un volantone con le offerte della settimana del supermercato, tutto va pensato in grande.
È come conseguenza di questa visione che i nostri prodotti sono arrivati in 10 paesi nel mondo, che il livello di fidelizzazione dei nostri clienti è al 90% o che il nostro fatturato si è triplicato.
Perché la grandezza è qualcosa che si crea guardando verso l'alto. Mai dall'alto in basso.

AGB | artigraficheBocciaspa | A DIFFERENT IMPRINTING.

ARTI GRAFICHE BOCCIA – SALERNO | ROMA | MILANO | PARIS | LONDON | LAUSANNE



CONTACT:
Via Tiberio Claudio Felice, 7 - 84131 Salerno (ITALY)
Tel. +39 089 303311 - Fax +39 089 771017
www.artigraficheboccia.com - info@artigraficheboccia.com

Grande potere, grandi responsabilità

Linux e il mondo Open Source sono diventati negli anni elementi indispensabili della società moderna in modi che il cittadino comune non immagina. Come vediamo nelle news, i sistemi aziendali usano sempre più spesso soluzioni aperte e chiunque abbia navigato su Internet è capitato per forza su siti ospitati da server basati su una distribuzione Linux perché il Pinguino è presente in due server su cinque. Gli ultimi mesi però non sono stati una pubblicità positiva per la sicurezza delle nostre distribuzioni preferite. Certo, Windows è notoriamente un colabrodo, quindi qualche problemino di Linux non deve fare gridare allo scandalo, ma è anche vero che i sistemi Open Source sono diventati i più diffusi nel mondo dei server Web anche per la loro resistenza ad attacchi e falle. Nel corso dell'ultimo anno sono stati parecchi i problemi evidenziati. Heartbleed ha spaventato gli utenti di tutto il mondo: una falla nel codice di OpenSSL ha esposto per anni dati sensibili a pirati e a organizzazioni governative. Dopodiché è arrivato Shellshock, il bug che permette a hacker di sfruttare la shell Bash per controllare sistemi in remoto. Infine è stato il momento di Turla, il virus che permette di lanciare comandi in remoto anche senza avere privilegi elevati. È inutile nascondersi: l'Open Source negli ultimi 12 mesi ha mostrato dei limiti di sicurezza. Noi di Linux Pro però la vediamo un po' diversamente da quelli che si sono lanciati nell'elogio di Windows e nell'attacco a Linux sfruttando queste vulnerabilità come esempio di come tutti i sistemi siano a rischio. L'Open Source è infatti una comunità che impara dai propri errori, e dagli errori degli altri, e si migliora. Linux non è diventata la piattaforma preferita per il Web (con quote di mercato in costante aumento) per caso. I problemi di Linux sono affrontati pubblicamente e velocemente, quelli di sistemi come Windows sono lasciati alla volontà di aziende il cui interesse è massimizzare i profitti. Sì, è vero, dobbiamo fare di più per rendere il mondo Open Source affidabile e resistente, ma questo perché ormai quandi si parla di sistemi critici Linux è diventato l'amico di cui fidarsi, il collega di lavoro che ha sempre la soluzione giusta, il fornitore che consegna sempre per tempo. E allora ben vengano le critiche: aiutano a migliorarsi e a mantenere alta l'attenzione. A differenza dei sistemi chiusi, il limite per l'Open Source è davvero il cielo...

La redazione di Linux Pro





CONTATTI

Domande alla redazione: **redazione@linuxpro.it**
Abbonamenti: **abbonamenti@myabb.it**
Arretrati: **arretrati@linuxpro.it**
Problemi con il DVD: **aiutocd@sprea.it**
Sito Web: **www.linuxpro.it**
Oppure inviate le vostre lettere a:
***Linux Pro*, Sprea Editori S.p.A.,**
Via Torino 51, 20063 Cernusco S/N
Telefono: 02.92432.1

LINUX PRO 150

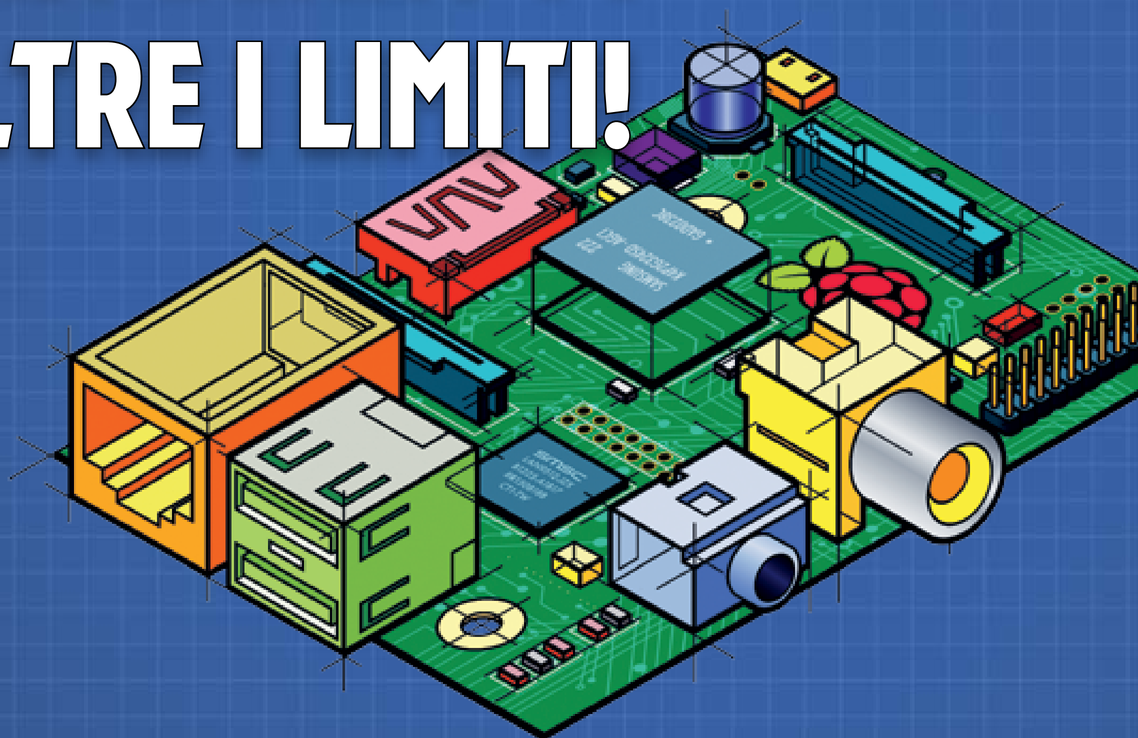
Sommario

LINUX PRO

Benvenuti nel centocinquantesimo numero di Linux Pro, la guida definitiva a Linux e al mondo Open Source

In primo piano

RASPBERRY PI OLTRE I LIMITI!





LINUX PRO

Sommario

IL PROSSIMO NUMERO ESCE IL 12 MARZO

IL DVD IN BREVE

LATO A

DISTRIBUZIONI
- Tango Studio 2.2 i386
- Tango Studio 2.2 x64

LATO B

DISTRIBUZIONI
- GhostBSD 4.0
- KXStudio 14.04



RIVISTA
- Dr. Geo 14.12
- Fame 0.96
- Guacamole 0.9.3 client
- Guacamole 0.9.3 server
- i3 4.8
- Fame
- QMMP 0.8.3
- Stumpwm 0.9.8
- TMux 1.9a
- Untangle

Quando trovi questo bollo negli articoli, cerca il software nel DVD



Prova la tua rivista anche in digitale

www.myabb.it/linuxpro











Newsdesk

Ogni mese tutte le novità dal mondo delle aziende e della comunità Open Source

Se hai news da segnalarci o dei commenti scrivi a newsdesk@linuxpro.it

Linux

Una crescita inarrestabile

Se scendessimo per strada e chiedessimo ai passanti se hanno mai usato Linux, molto probabilmente la stragrande maggioranza ci risponderebbe di no. Eppure, oggi più che mai, il sistema operativo del Pinguino fa parte del nostro quotidiano, magari a nostra insaputa, e questa tendenza pare davvero inarrestabile. Ne è un chiaro esempio quanto emerso dal CES di Las Vegas dello scorso gennaio, durante il quale sono stati presentati gli ultimi modelli di televisori Smart Ultra HD, i cui sistemi operativi sono quasi tutti basati su Android che, come sappiamo, è figlio di Linux. Quindi, anche buona parte dei nostri smartphone deve a quel kernel il proprio funzionamento. Tuttavia, sebbene questi due fronti già dovrebbero darci la misura del peso di Linux a livello mondiale e del suo continuo espandersi nei settori emergenti, non sono che la punta dell'iceberg. Infatti, se andiamo a dare un'occhiata a settori meno appariscenti, la storia non cambia, anzi è addirittura più chiara. Cominciamo dai supercomputer, quelle macchine nate per svolgere quantità impressionanti di calcoli in pochi istanti, appannaggio di grandi realtà dedite, per esempio, alla ricerca come la NASA. La classifica, stilata lo scorso novembre, dei 500 apparecchi più potenti del mondo vede una distribuzione di Linux presente su ben 485 di essi, pari al 97%! Di fronte a queste cifre, Windows e Android sembrano dei nani nei loro rispettivi ambiti. Tra l'altro, il Pinguino si trova proprio nel supercomputer che si è classificato primo, con i suoi 33.86 petaflop/s di potenza di calcolo, il cinese **Tianhe-2**. Sebbene nei desktop e nei laptop i sistemi operativi basati su Linux, come Ubuntu e Mint, non la facciano ancora da padrone, è innegabile che stiano erodendo fette di mercato, soprattutto a scapito di Microsoft. Ma è nell'universo dei server dove questa erosione diventa evidente. Infatti, secondo l'Enterprise End User Trends Report del 2014, solo negli ultimi quattro anni Windows ha perso in questo settore ben 9 punti. In pratica, dal 2011 al 2014, è sceso dal 45% al 36% del mercato. In controtendenza Linux che è salito di 12 punti, toccando il 79%. Teniamo presente che queste statistiche vengono redatte tenendo conto di aziende con un fatturato superiore ai 500 milioni di dollari o con più di 500 dipendenti, quindi stiamo parlando di corazzate come Morgan Stanley, Goldman Sachs e Deutsche Bank. Anche il cloud si avvicina a queste percentuali, visto che il 75% delle aziende dichiara di usare Linux come piattaforma principale, mentre Windows si attesta al 24%. Ma queste cifre sono solo una fotografia statica della situazione. Il trend è ben più impressionante, se pensiamo che la percentuale delle aziende intenzionate a passare a Linux nel 2015 è pari all'82%. Quando poi rivolgiamo la nostra attenzione agli sviluppatori di applicazioni, allora il quadro diventa ancora più definitivo, poiché quattro quinti di essi usano strumenti Open Source per svolgere il loro lavoro. Tirando le somme, non possiamo che constatare con soddisfazione come gran parte dell'universo informatico abbia ormai la prua rivolta verso questa nuova concezione di software dove il Pinguino la fa ovviamente da padrone assoluto. LXP



› **Sono basati su Linux anche i sistemi operativi degli apparecchi televisivi di ultimissima generazione, come gli Ultra HD, delle principali marche mondiali**



› **Attualmente, dei 500 supercomputer più veloci al mondo, ben 485 funzionano con una versione di Linux. Tra questi troviamo il cinese Tianhe-2, che è considerato il più potente**

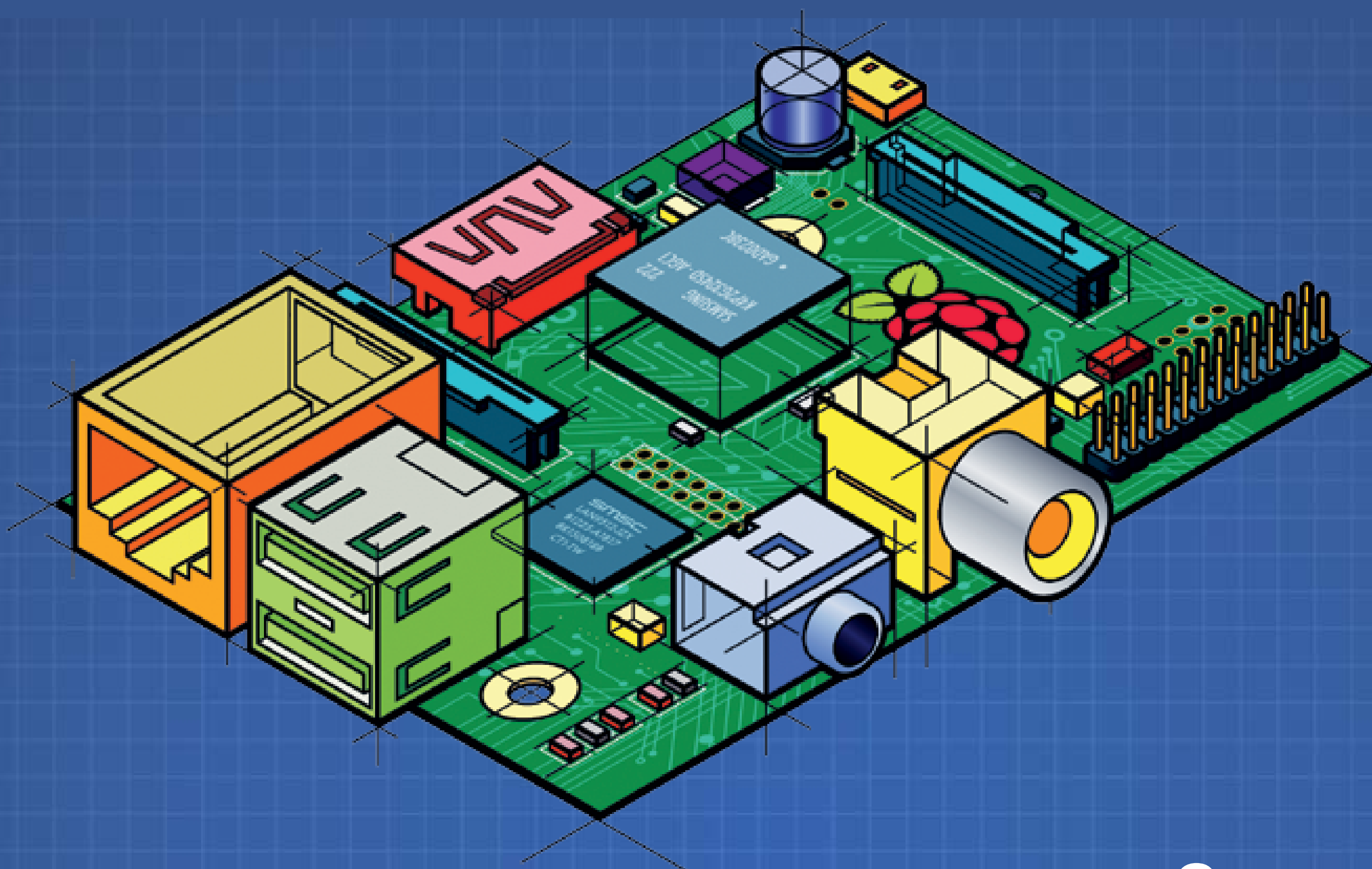
GRANDI NOVITÀ!





PRESTO IN EDICOLA

Sprea editori



Raspberry Pi oltre i limiti!

Ecco come padroneggiare al meglio gli intimi segreti della fantastica scheda Raspberry Pi

La Raspberry Pi fu inizialmente concepita come strumento didattico, un computer senza fronzoli per favorire i bambini che desideravano imparare a programmare ma che avevano difficoltà a reperire il danaro necessario per un PC classico. Ciononostante, la schedina piacque molto a "modder" e "hacker", che iniziarono a impiegarla in modo creativo, rendendola perciò usabile da una vasta platea. Non sembri quindi sorprendente se vi diciamo che armeggiare con la RPi è un bel modo per imparare a programmare e a usare GNU/Linux. E non siamo i soli a pensarlo: i nuovi curriculum degli scolari del Regno Unito (terra patria della Raspberry Pi) prevedono espressamente la conoscenza di aspetti informatici ormai entrati nella vita di tutti i giorni e che quindi devono essere parte del bagaglio culturale dagli uomini di domani. Vediamoli brevemente... Fase 1: i bimbi tra i 5 e i 7 anni sono chiamati a scrivere ed eseguire semplici programmini su differenti apparati, come tablet o la Raspberry Pi. Fase 2: i più grandicelli tra i 7 e gli 11 anni fanno la conoscenza di Scratch, il giocoso linguaggio di programmazione grafico installato di default su Raspbian, la versione di Debian ottimizzata per la Raspberry Pi. Fase 3: nella età tra gli 11 e i 14 anni gli studenti passano a linguaggi di programmazione testuale come il Python per poter controllare direttamente l'hardware. La RPi è una piattaforma perfettamente adatta a questo tipo di attività perché la presenza del bus di I/O permette il controllo delle dozzine di schede di espansione disponibili, una su tutte Pibrella. Infine, nella Fase 4, i ragazzi tra i 14 e i 16 anni possono usare tali eccellenti schede aggiuntive per sviluppare ulteriormente le ormai acquisite capacità programmatorie tramite la realizzazione di progetti propri. In questo articolo vi forniremo alcuni suggerimenti su come sfruttare al meglio la vostra Raspberry Pi, semplicemente assicuratevi di averne una a disposizione. Usate NOOB per preparare una schedina SD: scaricate l'immagine, scompattatela, riversate il risultato sulla SD ed eccovi pronti per partire. L'insieme dei trucchi che andremo a illustrare coprono praticamente le necessità di tutti i giorni, col grosso vantaggio di poter essere implementati da chiunque, indipendentemente dal grado di preparazione tecnica, senza contare che seguendo passo-passo le istruzioni acquisirete quei "trucchi del mestiere" che eventualmente vi mancano.

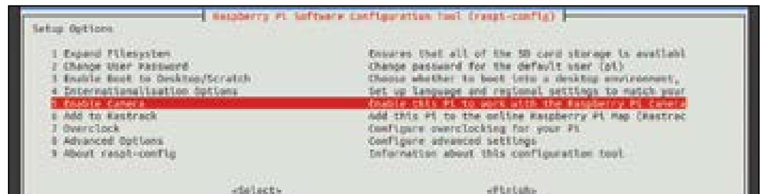
Competenze: Sicurezza, API Twitter

Hack #1: Inviare le foto su Twitter

La messa in commercio del "Raspberry Pi camera module" ha aperto un mondo di opportunità: chi già usava ingombranti Webcam USB ora ha a disposizione un minuscolo sensore ottico capace di riprese Full HD, ideale per attività come la sorveglianza. Per esempio, si possono inviare gli scatti eseguiti direttamente a un account Twitter: vediamo come. Cominciamo con il connettere il sensore al CSI (*Camera Sensor Interface*): è posizionato vicino alla porta LAN. Sollevate delicatamente lo "spinotto", inserite il cavo a "nastro" nella fessura facendo attenzione che i contatti argentati siano sul lato opposto rispetto al connettore Ethernet: rammentate di non premere con eccessiva forza il cavo all'interno del connettore in quanto potreste danneggiarlo. Ora, mantenendo il nastro nella posizione attuale, con l'altra mano premete verso il basso lo spinotto della CSI in modo da bloccare il nastro. Con l'hardware al suo posto è ora la volta di configurare il software. Avviate Raspbian e poi lanciare il tool di configurazione con **sudo raspi-config**, scorrete le opzioni verso il basso fino a raggiungere **Enable Camera**: attivatela e poi riavviate la RPi. Alla ripartenza sarete in grado di usare **raspistill** e **raspivid**, due comandi ben commentati in grado di catturare immagini il primo e filmati il secondo. Per scattare foto solo al cambio del panorama (il noto *capture motion*) la comunità ha realizzato uno script Python che fa parte della **Python Imaging Library** che, come fa intendere il nome, è una libreria dedicata alla manipolazione e analisi delle immagini. Cominciamo con l'istallazione tramite **sudo apt-get install python-imaging-tk**, dopodiché creiamo una directory di lavoro all'interno della propria home con **mkdir ~/picam**. Ora bisogna scaricare lo script con **wget -c http://pastebin.com/raw.php?i=yH7JHz9w -O picam.py** e renderlo eseguibile tramite **chmod +x picam.py**. Lanciando lo script con **./picam.py**, il LED rosso sulla camera RPi si accenderà, segnalando l'inizio delle riprese a bassa risoluzione. Il sistema si accorgerà degli oggetti in movimento confrontando la sequenza degli scatti: se differiscono, significa che qualcosa è cambiato di fronte alla telecamera e l'immagine verrà registrata, ma ad alta risoluzione.



› La distro Kali, specializzata in "penetration testing", è disponibile per Raspberry Pi



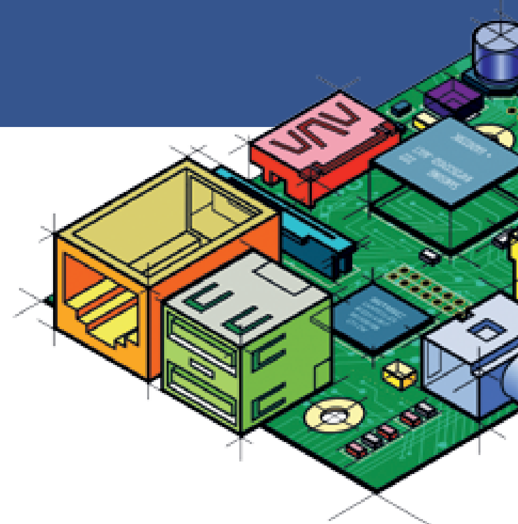
› Pesando solo 9 grammi, il modulo camera della Raspberry Pi può catturare immagini alla risoluzione di 2,592x1,944 tanto bene quanto un video in Full HD 1080p a 30 FPS oppure 720p a 60 FPS

Lo script è molto efficiente: archivia gli scatti ad alta risoluzione ed elimina quelli temporanei a bassa risoluzione, ottimizzando lo spazio su disco e l'utilizzo della CPU. Per automatizzare l'uso del programma, serve uno script che lo lanci alla partenza del sistema e che lo fermi in fase di spegnimento: ancora una volta, la comunità ha realizzato quanto serve, a voi basterà procurarvelo con **wget http://pastebin.com/raw.php?i=AfqbjQrb -O picam_init** e spostarlo nella posizione corretta con **sudo mv ~/picam_init /etc/init.d/picam** per poi renderlo eseguibile con **sudo chmod +x /etc/init.d/picam**. Per completare l'opera, avvisate il sistema che un nuovo script deve essere gestito e fatelo con **sudo update-rc.d picam defaults**. In tal modo, accensione/spegnimento del modulo camera saranno armonici con quello della Raspberry Pi. Naturalmente potete intervenire manualmente perché, come tutti gli altri demoni, anche picam può essere forzato dall'utente: per esempio, **/etc/init.d/picam** stop fermerà lo script e **/etc/init.d/picam start** lo avvierà.

Postare su Twitter

Ora attiveremo un nuovo account Twitter e su di esso posteremo foto tramite la Raspberry Pi. Assicuratevi che l'account sia privato e si cominci con l'installazione del software di base per la gestione delle librerie con **sudo apt-get install python-pip**. Ora è la volta di Twython: un "wrapper" Python per la gestione delle API di Twitter con **sudo pip install twython**. Si noti che per poter utilizzare Twython è necessario disporre su Twitter di un account per sviluppatori, per cui dirigetevi su **https://apps.twitter.com** e provvedete a registrare il vostro profilo: nella pagina che si apre selezionate **Create New App** e nello spazio disponibile inserite nome e descrizione. Saltate la parte relativa a **Callback URL field**, scorrete verso il basso e create l'app: poiché essa ha inizialmente solo i permessi di lettura, passate alla linguetta **Permissions** e attivate l'opzione **Read and Write**. Tornate alla linguetta **API Keys** e selezionate **Create My Access Token button**. Prendete nota delle opzioni presenti nella pagina, cioè **API Key**, **API Secret**, **Access Token** e **Access Token Secret variables**, quindi scaricate lo script picam modificato con **https://raw.githubusercontent.com/ghalfacree/bash-scripts/master/picamera-security.py > picam.py**. Con un editor di testo, aprite il file **picam.py** e inserite i 4 valori ricavati da Twitter nello spazio predisposto in cima allo script. Chiudete il file e rendetelo eseguibile con **chmod +x picam.py**. Il lavoro è completato: ora quando lo script è in esecuzione e rileva movimento, oltre a catturare l'immagine e archiviarla sulla schedina SD, provvederà a inviarla al vostro account Twitter privato, dove potrà essere visualizzata nel relativo **Twitter feed**.

»



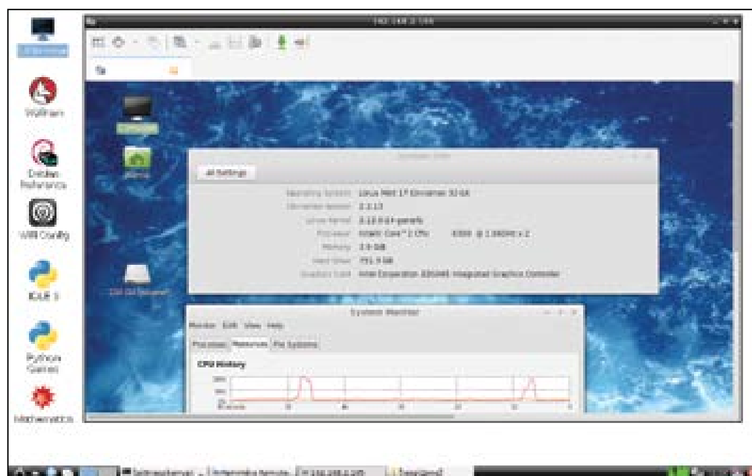
Competenze: Reti, gestione dati, interoperabilità

Hack #2: Condivisione file tramite un server Samba

La possibilità di accedere e condividere da altri PC i file presenti sulla Raspberry Pi è invero parecchio utile: per esempio, se la usate come una stazione di download perennemente attiva, avrete l'esigenza di accedere ai file scaricati, che siano presenti sulla SD o su un disco USB. Con il software Samba, che altro non è che una reimplementazione Libera del protocollo di rete SMB/CIFS, sarete in grado di usare la Raspberry Pi come se fosse un NAS (*Network Attached Storage*). Primo passo è l'installazione del software di base con **sudo apt-get install samba samba-common-bin**, dopodiché collegate alla RPI il disco USB che verrà automaticamente riconosciuto e montato nella directory **/media**: da adesso si dà per inteso che il disco esterno sia visto come **/media/usb**. Ora bisogna configurare Samba in modo che possa rendere accessibili i contenuti del disco: per farlo, aggiungete a Samba un utente chiamato 'pi' con **sudo smbpasswd -a pi** digitando di seguito la password che gli si vorrà associare. Ora va personalizzato il file di configurazione (**/etc/samba/smb.conf**) tramite un editor di testi. Se si desidera accedere alla Rpi da una macchina Windows, bisogna cercare la direttiva workgroup = WORKGROUP all'inizio del file **smb.conf** e cambiare il valore con un nome del proprio gruppo di lavoro Windows. Scendendo nel file, trovate la direttiva **# security = user** e attivatela rimuovendo il **#**: in questo modo abiliterete i controlli di sicurezza. Scorrete ancora verso il basso e alla fine del file aggiungete queste linee:

```
[USB]
path = /media/usb
comment = USB NAS Drive
valid users = pi
writeable = yes
browseable = yes
create mask = 0777
public = yes
```

Salvate le modifiche e riavviate il demone con **sudo /etc/init.d/samba restart.** Fatto: ora potrete accedere in lettura e scrittura ai file contenuti sul disco USB connesso alla RPi da tutti i PC presenti nella vostra rete locale.



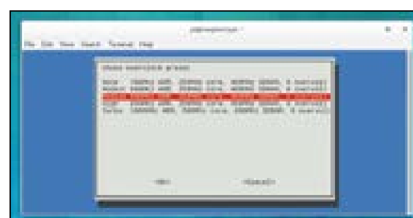
› **Remmina ha una interfaccia utente molto gradevole che scorre automaticamente quando il mouse raggiunge i limiti della finestra**

Hack #3: Raspberry Pi come 'thin client'

Con *thin client* (letteralmente 'client sottile') ci si riferisce a un computer dalle limitate capacità prestazionali che si appoggia a un sistema più potente che esegue tutto il lavoro vero e proprio per poi inoltrare al TC i risultati da visualizzare. Grazie ai bassi consumi e alla assenza di noiose ventole, la RPi è perfetta per tale scopo, affidandosi al protocollo di desktop remoto per comunicare col sistema principale

Overclockare la Pi

Ancorché 700 MHz sia una velocità sufficiente per permettere al processore di eseguire tutte le attività esposte in questo articolo, dopo un po' desidererete portare ai limiti le capacita della RPi. La bella notizia è che potete farlo!
Il SOC BCM2835 può essere spinto a velocità superiori, ma al prezzo di maggiori consumi energetici ed elevato calore prodotto... e mettete in conto anche una possibile instabilità di funzionamento e relativo "crash" di sistema. Nonostante sia possibile intervenire manualmente, la strada più sicura è attivare la procedura di *overclock* tramite il tool **raspi-config** il cui uso non invalida la garanzia: le impostazioni di raspi-config sono note per essere stabili sulla maggioranza delle schede. Per cambiare la frequenza di funzionamento, lanciate il tool, scorrete le opzioni verso il basso fino alla voce **Overclock** e confermate che volete proseguire. Verrà mostrata una lista di frequenze pre-impostate tra le quali scegliere. Dopo che avrete deciso, verrà eseguito uno riavvio automatico della RPi per attivare la nuova velocità di clock.

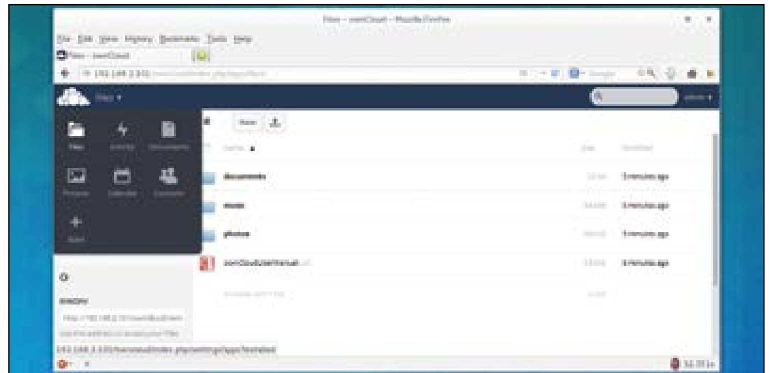


› **Se la RPi fallisce il boot a causa dell'eccessivo overclock, riavviate tenendo premuto il tasto Shift: verranno così reimpostati i valori di default**

che è implicito debba essere un PC/server dalle prestazioni più che buone. L'unità che ci fa da server usa Ubuntu e ha preinstallato **Vino remote desktop**, accessibile dalla RPi tramite il programma **Desktop Sharing**. Vino però ha l'opzione di accesso remoto disabilitata di default: per attivarla, lanciate il programma e selezionate **Allow other user to view your desktop** che sblocca la seconda opzione abilitando la quale, potrete permettere il controllo remoto del vostro server Ubuntu. Per sicurezza, attivate anche la richiesta di credenziali d'accesso nella sezione **Security** per forzare l'uso di password e quindi consentire solo accessi autorizzati. Anche se potreste voler saltare la gestione delle password disabilitando l'opzione relativa, consigliamo di attivarla imponendone una molto robusta nella relativa riga di inserimento. Completata questa fase, selezionate **Close** per salvare i cambiamenti. Nel caso usiate altro rispetto a Ubuntu, potete installare un server VNC come il già citato Vino oppure **Krfb**. Usando Vino su Ubuntu, dovrete fare un'ulteriore configurazione in quanto il programma ha impostato di default la richiesta di criptazione ma supporta solo vecchi protocolli che non sono più molto diffusi. Aprite dunque un Terminale su server Ubuntu e modificate le impostazioni di sicurezza di Vino con **gsettings set org.gnome.Vino require-encryption false**. Verificate che la connessione remota dalla RPi funzioni, quindi rendete permanenti le modifiche sul server installando **dconf-editor** con **sudo apt-get install dconf-editor**, lanciate il programma, portatevi su **org ➟ gnome ➟ desktop ➟ remote-access** e deselezionate l'opzione **require-encryption**. Ora passiamo alla RPi, sulla quale è meglio usare un software leggero come Remmina, installabile con **sudo apt-get install remmina**. Lanciate il programma (lo trovate nel menu **Internet**) e fate click su **new icon** per configurare la connessione al server. Nella finestra che si apre, date un nome alla connessione e selezionate VNC dal menu a tendina dei protocolli disponibili e completate inserendo l'indirizzo IP del server nel campo **Server**. Potete migliorare l'aspetto estetico della connessione alzando la profondità di colore e migliorando la qualità delle immagini selezionando i livelli voluti nei relativi menu a tendina. Selezionate **Connect** e Remmina attiverà la comunicazione col Server Ubuntu, richiedendovi la password che avevate impostato in fase di configurazione, permettendovi così di prendere il controllo del potente server dalla debole RPi.

Hack #4: Attivate la vostra nuvola con OwnCloud

Se volete che i vostri dati siano universalmente raggiungibili, non avete per forza bisogno di sprecare denaro in servizi che piazzano i vostro dati all'estero, sottraendoli alla giurisdizione del vostro Paese e con tutte le possibili conseguenze legali del caso. Invece potreste decidere di investire risorse finanziarie per procurarvi una RPi e un generoso disco USB e attivare la vostra 'nuvola', protetta e personale tramite OwnCloud, un software che permette di aggiornare e condividere file da e con qualunque risorsa connessa a Internet. Se poi la sicurezza è una condizione indispensabile, la possibilità di crittografare i dati vi farà dormire sonni meno agitati. OwnCloud può gestire file di varia natura e ha incorporate già alcune funzioni di base come l'album fotografico e il riproduttore multimediale. Una caratteristica molto importante è il *versioning*, cioè la possibilità di avere varie versioni dei file caricati, in modo da poter 'tornare indietro' nel caso di errori e recuperare la versione di 'n' giorni fa di quell'importante documento che state preparando. Come su tutti i servizi di archiviazione in Rete, avete la possibilità accedere a OwnCloud tramite un navigatore Web generico (Firefox, Chromium, ecc.) oppure tramite un client dedicato disponibile per Windows, Mac OS X e GNU/Linux, oltre che per sistemi che usano iOS o Android. Owncloud necessita del server Web Apache e di un database. MySQL è il più usato ma per questo lavoro ci



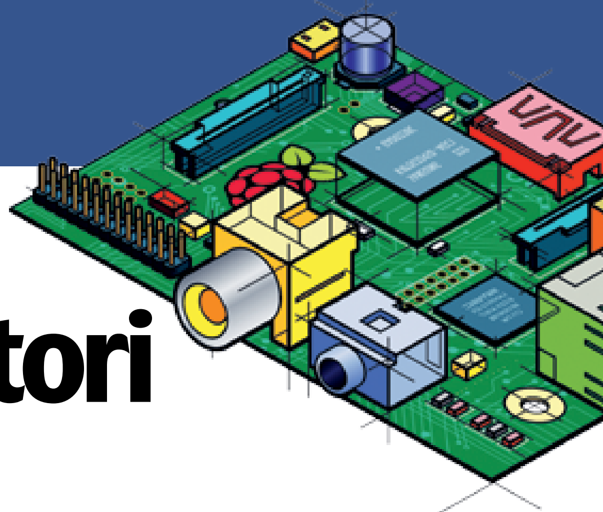
› **OwnCloud 7 è un significativo passo avanti rispetto alle precedenti versioni, assicurando un flusso di lavoro semplificato**

appoggeremo al più leggero SQLite, maggiormente in sintonia con le limitate capacità computazionali della Raspberry Pi. Installate i componenti necessari con **sudo apt-get install apache2 php5 php5-gd php5-sqlite curl libcurl3 php5-curl**, poi andate sul sito **https://owncloud.org** e scaricate l'archivio tarball dell'ultima versione disponibile (7.0.2 al momento della scrittura di questo articolo), scompattatela con **tar xjvf owncloud-7.0.2.tar.bz2** dopodiché spostate la directory risultante nella radice del vostro server Web con **sudo mv owncloud /var/www**. Ora assicuratevi che i nuovi file abbiamo i permessi coerenti con la loro nuova posizione, tramite il comando **sudo chown -R www-data:www-data /var/www/owncloud**. Avete bisogno di abilitare alcuni specifici moduli di Apache

> **"Se volete rendere universale l'accesso ai vostri dati, procuratevi una Pi, un disco USB e installate la vostra 'nuvola' personale"**

per far sì che OwnCloud funzioni correttamente. Digitate **sudo a2enmod headers rewrite env**, dopodiché attivate le modifiche con **sudo service apache2 restart**. La configurazione di OwnCloud si fa tramite un normale browser Web: digitate nella barra degli indirizzi **localhost/owncloud**. Poiché è la prima volta che si lancia OwnCloud, vi verrà richiesto di attivare un account con diritti amministrativi. Ora potete connettervi con diritti di amministratore e caricare i file che volete condividere, ma prima di farlo, occorre sistemare la configurazione di PHP in quanto di default il limite di upload è posto a un massimo di 2 MB. Con un editor di testi aprite il file **php.ini** che trovate in **/etc/php5/apache2**, cercate le variabili **upload_max_filesize** e **post_max_size variables** e modificate il loro valore a qualcosa come 500 MB o 1 GB. Ora potete interagire con il server OwnCloud tramite interfaccia Web, ma è anche possibile usare il protocollo WebDAV: in questo caso, aprite un qualunque File Manager, premete **Ctrl+L** per posizionarvi nella barra dei file e lì digitate: **dav://<indierizzo_del_server_OwnCloud>/owncloud/remote.php/webdav**. Dopo esservi autenticati, lo spazio-file di OwnCloud sarà a disposizione come una normale directory del vostro PC. Per condividere i file caricati tramite l'interfaccia Web, andate nella sezione **Files**: alla destra del file di vostro interesse è presente una lista di opzioni tra le quali **Share**, che vi permetterà di selezionare con chi condividerlo, oltre alla possibilità di permetterne addirittura la modifica e cancellazione. È anche consentito condividere un dato con chi non è registrato sul vostro server: quando sceglieste **Share**, si attiva il link che potete condividere con chiunque su Internet. Si può anche impostare una password e una data di scadenza del link.

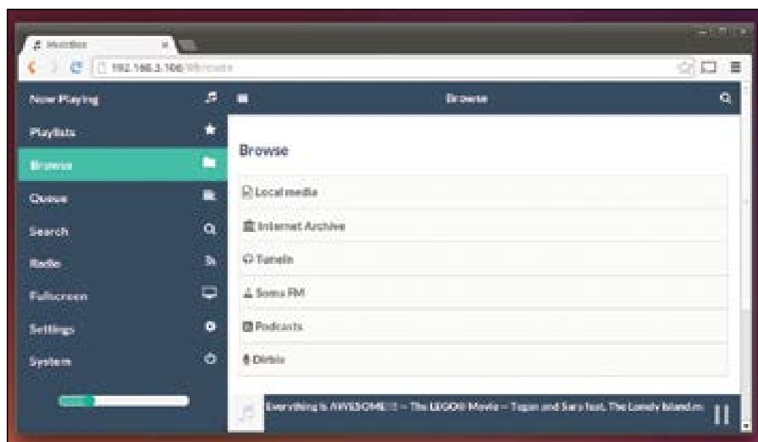
»



Competenze: streaming, emulatori

Hack #5: Musica dal vivo direttamente da Internet

Abbiamo musica ovunque: oltre alla collezione di musica libera sui nostri dischi fissi, probabilmente abbiamo centinaia di titoli su Spotify o Google Play. Bene, potete avere tutto-in-uno grazie alla distribuzione **PiMusicBox** che trasforma la Raspberry Pi nell'ultima parola in fatto di server musicali. La PiMusicBox può riprodurre file audio sull'uscita analogica tanto bene quanto sulla HDMI o via USB: è sufficiente installare la distro, connettere l'uscita audio che più vi aggrada e via che si parte! Ovviamente il sistema è controllabile da remoto tramite un normale PC o un terminale Android. Si comincia scaricando l'immagine compressa di PiMusicBox da **www.pimusicbox.com**: scompattatela e riversatela sulla SD col solito **sudo dd if=musicbox0.5.img of=/dev/sdd**, avendo cura di rimpiazzare **/dev/sdd** con la reale corrispondenza della vostra scheda SD. Se usate la porta LAN per connettere la RPi a Internet, potete tranquillamente usare la SD appena creata per fare il boot. Nel caso invece utilizziate una connessione senza fili, dovrete editare manualmente il file di configurazione relativo e impostare manualmente i parametri di collegamento al vostro router wireless: ecco come fare. Tramite un altro PC Linux accedete alla SD cercando il file **settings.ini** che aprirete tramite un editor di testi: Noterete le due variabili **WIFI_NETWORK** e **WIFI_PASSWORD** che avrete cura di inizializzare con nome e password della vostra rete Wi-Fi. Attenzione: PiMusicBox lavora solo su reti che usano il protocollo WPA2. Finita la configurazione e rimessa la SD al suo posto, accendete la Rpi: come prima operazione si attiverà il ridimensionamento del filesystem allo scopo di occupare tutto lo spazio disponibile, dopodiché Linux si riavvierà automaticamente. Se avete connesso un monitor, potrete seguire visivamente tutte le fasi del processo, altrimenti calcolate un paio di minuti dall'accensione e poi da un altro PC usate un browser per accedere alla Raspberry Pi tramite **http://musicbox.local**



› **PiMusicBox si basa su server musicale Mopidy, scritto in Python**

e se non dovesse funzionare, usate direttamente l'indirizzo IP. L'interfaccia di PiMusicBox è piuttosto spoglia poiché non avete ancora configurato nessuna sorgente musicale: per farlo, nella barra di navigazione sulla sinistra selezionate **Setting** e verrete portati nella pagina dove potrete abilitare e configurare i vari servizi di streaming a pagamento come Spotify e Google Music oppure quelli gratuiti come The Internet Archive, Soma FM e altri. Potete controllare anche altre impostazioni in questa pagina: per esempio, le impostazioni audio vi permettono di selezionare quale uscita sonora utilizzare. Siccome Samba è già installato e configurato su PiMusicBox, potrete accedervi facilmente dagli altri sistemi presenti sulla vostra rete locale. La directory condivisa si chiama **music**: qualunque cosa inserirete da remoto in tale spazio verrà automaticamente ricopiata sulla SD. Ogni volta che riavvierete la RPi, si avvierà una scansione automatica per rilevare eventuali nuovi file musicali che potrete riprodurre tramite l'interfaccia Web. Concludiamo dicendo che potete suonare musica tramite qualunque software che supporti l'MPD (*Music Player Daemon*) come per esempio MPDroid per Android: per connettervi, lanciate il programma e nella fase guidata di configurazione inserite l'indirizzo IP della RPi nel campo **hostname** e poi salvate.

Hack #6: Radio FM con la Raspberry Pi

Ricevere musica sulla RPi è una cosa, ma se invece volessimo inviare musica dalla RPi ad altri sistemi? Che ne direste di gestire una vostra radio digitale? Oltre ai familiari connettori audio, Ethernet, USB e HDMI, la RPi dispone di una interfaccia concepita per connessioni dirette con altri chip e moduli elettronici. Stiamo parlando del GPIO (*General Purpose Input/Output*), quella serie di piedini disposti in una doppia fila di 26 elementi sul modello B, mentre sono 40 sul modello B+. Questi pin non sono di comprensione immediata, ma hanno il vantaggio di poter essere controllati via software: un gruppo di programmatori del Code Club ha scritto un programma che converte quei piedini in un emettitore di segnali radio in Modulazione di Frequenza. Per trasmettere un segnale FM sorprendentemente forte non dovrete fare altro che connettere un cavo-antenna al piedino 4 del bus GPIO. Anche senza antenna, il segnale è abbastanza buono per essere captato da qualunque ricevitore FM a patto che non sia troppo lontano. Accendete il dispositivo, andate a riga di comando, scaricate il codice del Code Club e scompattatelo:

```
wget http://omattos.com/pifm.tar.gz
mkdir ~/pifm
tar zxvf pifm.tar.gz -C ~/pifm
```

Il file compresso contiene sei file, non ne servono di più. Ora potete trasmettere il file di prova **sound.wav file con sudo ./pifm sound.wav 101.2**. Siate lesti a configurare la vostra radio ricevente sulla frequenza di 101.2 e sentirete la colonna sonora di Guerre Stellari. Potete ovviamente

cambiare la frequenza di trasmissione tra 88 MHz e 108 MHz semplicemente aggiungendo il valore voluto in coda al comando come il 101.2 nell'esempio. Potete inviare nell'etere qualunque file sonoro in formato WAV a patto che come parametri sia: 16-bit, 22.050 Hz e mono. Potrebbe sembrare una grossa limitazione ma non lo è, infatti potete rapidamente convertire qualunque file nel formato voluto grazie a **SoX**, l'ottimo programma di conversione audio disponibile in tutte le distro Linux. A riprova delle nostre parole, useremo un file MP3, convertendolo 'al volo' in WAV prima di mandarlo nell'etere. Installiamo il programma e le sue dipendenze con **sudo apt-get install sox libsox-fmt-all**. Ora facciamo la prova: ovviamente sostituirete 'pippo' con il nome del vostro file da convertire:

```
sox -t mp3 pippo.mp3 -t wav -r 22050 -c 1 - | sudo ./pifm - 101.2
```

La prima parte converte il file da MP3 in WAV, imponendo la campionatura a 22.050 Hz e usando un solo canale, dopodiché il file è inviato allo standard output (il 'meno' dopo l'1) al che viene raccolto dal simbolo di 'pipe' (|) che lo rimanda all'ingresso del comando **pifm**. L'unica differenza col precedente esempio è che invece di mandare in onda un file tramite il suo nome, lo abbiamo fatto fruttando lo "standard input". Se avete la radio ancora sintonizzata sui 101.2, dovreste sentire l'MP3. Potete fare cose veramente notevoli con SoX: per esempio si può prendere un flusso sonoro da Internet e rimandarlo in FM, ecco come fare per trasmettere un podocast TuxRadar

```
sox -t mp3 http://www.tuxradar.com/files/podcast/tuxradar_s06e02.mp3 -t wav -r 22050 -c 1 - | sudo ./pifm - 101.2
```

Hack #7: Emulare vecchi giochi

I giochi non sono sempre stati delle grandi opere d'arte grafica. Inizialmente i loro autori avevano poche risorse a disposizione, quindi al posto degli effetti speciali visivi si affidavano a storie intriganti per mantenere alto l'interesse dei giocatori. Questo è il motivo per cui vecchi giochi dalla grafica rudimentale sono tuttora noti tra i giocatori di tutte le età: non c'è nulla di meglio di un rompicapo di Monkey Island ancorché riproposto su una piattaforma moderna come Steam o Live. Il modo più facile per cominciare a giocare i vecchi titoli sulla Raspberry Pi è procurarsi **RetroPie**, che contiene in blocco diversi emulatori. Due alternative: installare il programma su Raspbian oppure scegliere la distro dedicata. Nel primo caso, procuratevi lo script e le varie dipendenze con **sudo apt-get install git dialog**. Poi scaricate l'ultima versione di RetroPie con:

```
git clone git://github.com/petrockblog/RetroPie-Setup.git
```

Ora entrate nella directory e lanciate lo script con **cd RetroPie-Setup && sudo ./retropie_setup.sh**. Il programma scaricherà le dipendenze mancanti e poi presenterà un menu di scelta con due opzioni: la prima scarica e installa i binari precompilati, la seconda acquisisce i sorgenti più recenti e li compila sulla RPi. L'ultima opzione presuppone che siate disposti ad aspettare parecchie ore la fine della compilazione. Se siete impazienti, potete procurarvi un'immagine già pronta e riversarla sulla memoria SD tramite il solito dd. RetroPie usa una interfaccia grafica chiamata **EmulationStation** che permette la gestione di vari emulatori: il 'frontend' parte in automatico se avete scelto l'immagine già pronta, mentre se avete installato RetroPie su una distro già esistente, dovrete lanciare il programma digitando **emulationstation** in un terminale. Tramite il programma potrete configurare il vostro controller di gioco USB. Ovviamente per giocare dovete possedere le ROM: potete ricavarle dall'hardware originale del gioco oppure più facilmente da qualche sito di 'abbandonware' presente in Rete. Alcune aziende hanno 'liberato' i propri programmi come per esempio iD software l'ha fatto con **Doom**, rendendolo di pubblico dominio. Dopo averle trovate, copiate le ROM nell'appropriata posizione del relativo emulatore, all'interno della directory **roms/** sulla scheda SD. »



› **EmulationStation mostrerà solamente gli emulatori che dispongono di ROM aggiunte**

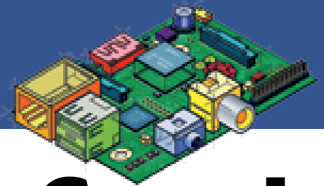
> **"Prerequisito fondamentale per emulare un gioco è possedere legalmente la relativa ROM..."**

Donate le risorse superflue

Se non usate la RPi in modo continuativo, potreste donare i momenti di pausa impiegando tali risorse in attività meritevoli. **BOINC** è un servizio di aggregazione di risorse su base volontaria e la potenza elaborativa accumulata viene impiegata in vari compiti, dalla ricerca medica all'elaborazione dei segnali captati dai radio-telescopi alla ricerca di segnali intelligenti alieni. Prima di installare BOINC, usate **raspi-config** per ottimizzare l'uso della memoria: andate in **Advanced Options** ➠ **Memory Split** e mettete al minimo (16 MB) la RAM da dedicare alla grafica, poi installate BOINC con **sudo apt-get install boinc-manager boinc-client**. Ora lanciate il programma dal menu **Others**. Vi verrà chiesto di scegliere uno dei 30 progetti attualmente in attività: scegliete con attenzione perché alcuni non sono adatti a hardware limitato come quello della RPi. Attività a basso consumo di memoria sono da preferirsi, per esempio potreste decidere di aggregarvi al gruppo che sta tentando di confutare la Congettura di Collatz (**https://it.wikipedia.org/wiki/Congettura_di_Collatz**).



› **Per configurare BOINC prima di tutto abilitate 'Advance View' e poi andate in Tools ➠ 'Computing preferences'**



Competenze: Python, eSpeak

Hack #8: Fate parlare la vostra RPi

I software di sintetizzazione vocale sono sempre molto popolari per quella parvenza di vita che danno alle macchine tramite la parola. Più prosaicamente, il loro valore è evidente in quando rendono fruibile un computer anche a coloro che hanno problemi di vista. Potete usare la Raspberry Pi per la lettura di testi grazie alla disponibilità della potente libreria **eSpeak** presente nei repository della Raspbian. C'è anche un modulo che permette di usare eSpeak tramite Python permettendovi di automatizzare tutta una serie di attività. Aprite un terminale e lanciate **sudo apt-get install espeak python-espeak** per scaricare le librerie e i relativi moduli Python. L'uso di eSpeak è molto lineare: digitate **espeak "Hello! How are you doing, today?"** in un terminale e la libreria userà le sue parametrizzazioni di default per leggere il testo tra le doppie virgolette. Potete influenzare la lettura tramite un'ampia serie di opzioni a riga di comando: per esempio, **espeak -ven+f2 -s140 "Aren't you a little short for a storm-trooper"**, leggerà il testo con un ritmo più lento e con voce femminile. Tra le varie opzioni, eSpeak pronuncia con accento statunitense o scozzese. Oltre all'inglese, sono presenti dozzine di altre lingue tra cui l'italiano: digitando **espeak --voices** verranno elencate tutte le possibilità.

Il modulo Python eSpeak è comodo per rapide conversioni da testo a voce. Attivate la console interattiva digitando **python** alla riga di comando e poi inserite il seguente testo:

```
from espeak import espeak
espeak.synth("How ya doin'?")
```

Ora usate la libreria eSpeak all'interno di uno script Python per stampare e leggere il nome dei vostri 'followers' su Twitter. Seguite le istruzioni del primo "trucco" per attivare le API di Twitter e procuratevi i token di accesso che provvederete a inserire tra gli apici singoli del codice seguente:

```
import time
from twython import Twython
from espeak import espeak
api_token = ' '
api_secret = ' '
access_token = ' '
access_token_secret = ' '
twitter = Twython (api_token, api_secret, access_token,
name='geekybodhi',cursor=next_cursor)
for result in search['users']:
print result["name"]
their_name= result["name"]
espeak.synth(their_name)
time.sleep(2)
next_cursor = search["next_cursor"]
```

Il precedente codice vi connette al vostro account Twitter e recupera la lista degli iscritti, i cui nomi verranno separati tramite la tecnica del *cursoring* che converte la lista in singoli elementi che verranno puntati tramite il comando **next_cursor**. Il ciclo **For** passerà gli elementi dal primo all'ultimo e per ognuno di essi stampa a video il nome e lo immagazzina nella variabile **their_name** che verrà passata alla libreria eSpeak per la conversione da testo ad audio. Dopo una pausa di due secondi, il ciclo ricomincia dal nome successivo.

```
pi@raspberrypi: ~/jasper/client
pi@raspberrypi ~/jasper/client $ python populate.py
Welcome to the profile populator. If, at any step, you'd prefer not to enter the
 requested information, just hit 'Enter' with a blank field to continue.
First name: Mayank
Last name: Sharma

Jasper uses your Gmail to send notifications. Alternatively, you can skip this s
tep (or just fill in the email address if you want to receive email notification
s) and setup a Mailgun account, as at http://jasperproject.github.io/documentati
on/software/#mailgun.

Gmail address: geekybodhi@gmail.com
```

› **Potete personalizzare ed espandere Jasper aggiungendo comandi realizzati da voi**

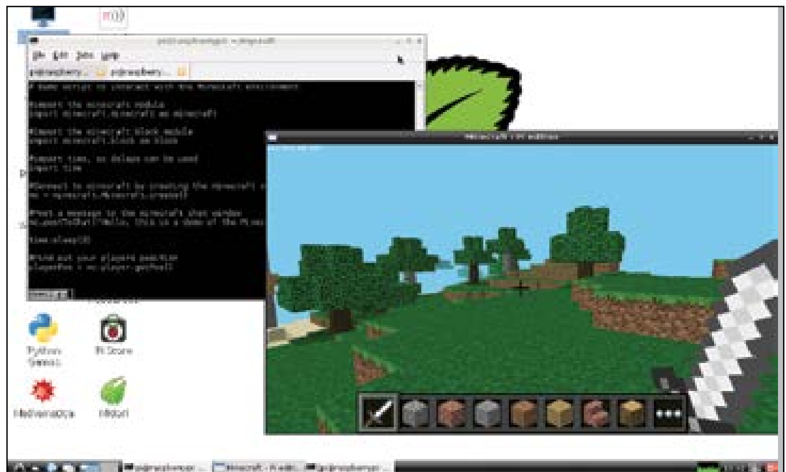
Hack #9: Controllare la RPi tramite la voce

Apple ha Siri, Google ha Google Now e Microsoft ha Cortana. Se pensate che la Raspberry Pi non abbia un proprio assistente vocale significa che non avete mai sentito parlare di **Jasper** che, a differenza dei nomi prima citati, è a sorgente aperto. Per sentire cosa dice Jasper, serve ovviamente connettere le casse o gli altoparlanti all'uscita audio e siccome l'hardware non dispone di ingresso audio, per cui bisogna ricorrere a un microfono USB supportato dalla RPi. Se vi sentite avventurosi potete provare a connettere una Webcam: alcuni modelli sono dotati di microfono. Va bene, diamo per scontato che vi siete dotati di una "periferica di ingresso sonoro" e andiamo avanti: dopo aver connesso il microfono assicuratevi che funzioni digitando **arecord –l**. Se riconosciuto, il microfono verrà citato per marca e modello. Anche se potete installare Jasper in una Raspbian già esistente, la strada raccomandata è quella di usare una immagine per SD già pronta. Procuratevi il file compresso dal sito Web **http://jasperproject.github.io** e scompattatelo con

```
tar zxvf jasper-disk-image.tar.gz.
```

Supponendo che la memoria SD sia vista come **/dev/sdd**, riversateci sopra l'immagine con **sudo dd if=jasper-disk-image.img of=/dev/sdd**. Completata la scrittura, spostate l'immagine sulla RPi e avviatela. Potete configurare Jasper da remoto tramite una connessione cablata SSH, ma se avete solo il Wi-Fi siete obbligati a connettere tastiera, monitor e abilitare la connessione senza fili tramite l'apposito programma. Utenza e password sono pi:raspberry, le stesse della Raspbian. Ok, il sistema è attivo e voi siete dentro tramite SSH: nella home directory della RPi scaricate il client Jasper con **git clone https://github.com/jasperproject/jasper-client. git jasper**. Aggiornate quindi con **sudo pip install upgrade setuptools**. Il prossimo comando serve a scaricare i componenti Python: **sudo pip install -r jasper/client/requirements.txt.** Infine impostate i diritti all'interno della home directory con **sudo chmod 777 -R** * e riavviate la RPi. Alla ripartenza, Jasper eseguirà lo script **boot.py** generando il file **languagemodel.lm** all'interno della directory **~/jasper/**. Assicuratevi che il file sia stato effettivamente creato: se così non fosse potete lanciare manualmente lo script boot.py con **~/jasper/boot/boot.py**. Una volta disponibili i modelli di linguaggio, create il profilo utente per Jasper in modo che impari a conoscere la

vostra pronuncia. Spostatevi nella directory **~/jasper/** e date il comando **python populate.py**. Lo script chiederà varie informazioni (nome, ecc.) e le userà per comunicare con voi e rispondere alle vostre domande in modo più accurato. Tenete presente che Jasper vi chiederà la password del vostro account di posta elettronica e la memorizzerà in modo leggibile: i dettagli li trovate all'interno del file **profile.yml**. Seguite la documentazione sul sito Web del progetto per integrare gli altri servizi come Facebook e Spotify. Riavviate la RPi dopo aver eseguito lo script di creazione profilo, dopodiché Jasper vi darà il benvenuto con la voce sintetizzata che avete precedentemente selezionato. Ora potete cominciare a interagire col vostro assistente semplicemente dicendo 'Jasper': il sistema risponderà con un segnale acustico, al che potete porre domande del tipo 'What's the time' o 'Do I have new email?'. Se Jasper vi sente, risponderà. In funzione della qualità del microfono e delle abilità oratorie del parlante, le prime conversazioni con Jasper potrebbero essere di qualità non soddisfacente.



› **Abbiamo già visto in passato come costruire e far sparare un cannone in Minecraft**

Hack #10: Minecraft per RPi

La versione di **Minecraft** per RPi è una versione ridotta della 'Pocket Edition' ma dotata di sufficienti caratteristiche per non limitare la vostra creatività. Potete esplorare mondi generati casualmente e usare la modalità 'costruzione' per dare sfogo alla vostra fantasia architetturale. La cosa migliore è la disponibilità delle API, accessibili tramite Python: potete realizzare i vostri script per muovere il personaggio tanto quanto per assemblare o distruggere blocchi. Tutto ciò vi consente di creare strutture complesse in pochissimi secondi, quando farlo a mano richiederebbe ore e ore di tedioso lavoro. Le API consentono anche la creazione di oggetti interattivi come orologi o portali di teletrasporto: imparerete a programmare divertendovi! Per installare la 'RPi Edition' andate su **pi.minecraft.net** e scaricate l'archivio compresso, dopodiché espandetelo nella vostra directory personale con **tar zxvf minecraft-pi.tar.gz:** il comando provvederà a creare la directory **~/mcpi** dove dovrete entrare e lanciare il gioco con **./minecraft-pi**. Se Minecraft non vi è familiare, sappiate che i movimenti sono gestiti coi soliti tasti **WASD**, i pulsanti dall'1 all'8 selezionano gli oggetti contenuti nella **Barra di Scelta Rapida** e lo **Spazio** fa saltare l'omino: infine, un doppio tocco sempre sulla **Barra spaziatrice** attiva/disattiva la modalità **volo**. Potete usare le API per connettervi a una sessione in esecuzione di Minecraft. Però, prima di farlo, è consigliato creare un copia della directory **/mcpi** per sicurezza: nel terminale digitate **mkdir ~/mycraft** per creare la directory e copiateci dentro i file con **cp -r ~/mcpi/api/python/mcpi ~/mycraft/minecraft.** Ora lavorerete solo nella directory **~/mycraft** così i file originali non verranno toccati e potrete "tornare indietro" in caso di danni. Mentre il gioco è in funzione, premete **Alt+Tab** per tornare al Terminale, aprite una nuova "tab" per e spostatevi nella directory mycraft. Aprite un editor di testi e create un file con **nano ~/mycraft/demo.py** che popolerete col testo seguente:

```
# Demo script to interact with the Minecraft environment
import minecraft.minecraft as minecraft
import minecraft.block as block
import time
#Connect to minecraft by creating the minecraft object
mc = minecraft.Minecraft.create()
#Post a message to the minecraft chat window
mc.postToChat ("Questa è una demo di Minecraft API.")
time.sleep(2)
playerPos = mc.player.getPos() #Find your player's position
#Change your players position
mc.postToChat("Andiamo 50 blocchi nel cielo!")
time.sleep(2)
mc.player.setPos(playerPos.x,playerPos.y + 50,playerPos.z)
# wait for 10 seconds while you fall back
time.sleep(10)
# - create a STONE block in front of you
playerPos = mc.player.getTilePos()
mc.setBlock(playerPos.x+1, playerPos.y+1, playerPos.z,
block.STONE)
time.sleep(5)
# - Now change that block into WOOD
mc.setBlock(playerPos.x+1, playerPos.y+1, playerPos.z,
block.WOOD_PLANKS)
time.sleep(5)
# - Now let's create a tower by stacking block
for top in range(0, 10):
mc.setBlock(playerPos.x+3, playerPos.y+top, playerPos.z,
block.STONE)
time.sleep(5)
# - Let's now teleport you to the top of the tower
mc.player.setPos(playerPos.x+1, playerPos.y+10, playerPos.z)
```

Salvate lo script e sempre con Minecraft in funzione, attivate lo script con **python ~mycraft/demo.py**. Se non avete commesso errori di digitazione, vedrete un messaggio di benvenuto così come impostato nella variabile **postToChat**. Dopodiché verranno acquisite le tre coordinate spaziali Z, Y e Z del personaggio (visibili in alto a sinistra sullo schermo) tramite **getPos** e archiviate nella variabile **playerPos**. Con **setPos** possiamo cambiare la posizione dell'omino, magari teletrasportandolo a mezz'aria: una volta ricaduto al suolo, con **setBlock** verrà creato un cubo di pietra (block.STONE) di fronte al personaggio e dopo 5 secondi la pietra si trasformerà in legno (block.WOOD_PLANKS). È ora il turno di un ciclo **For** che piazzerà poco lontano una colonna di 10 blocchi di pietra sulla cui sommità verrà posizionato l'omino, ultima azione prima di terminare lo script. Questo semplice demo è solo un esempio delle grandi opportunità di automazione rese possibili dalle API Python: scaricate questo bel documento PDF (**http://bit.ly/MinecraftPyCheatSheet**) che contiene tutta una serie di trucchi sull'utilizzo delle API, oltre naturalmente ad alcuni esempi chiarificatori su come impostare i vari parametri. Usare Python per controllare Minecraft è estremamente potente e offre un'elevata flessibilità che non abbiamo potuto approfondire in queste poche righe a nostra disposizione. Speriamo comunque di avervi incuriosito a sufficienza per stimolarvi a procurarvi i precedenti numeri di Linux Pro, dove troverete altri tutorial sull'uso di Python e Minecraft. LXP

LA NUOVA GENERAZIONE 1&1 CLOUD SERVER

Easy to use – ready to cloud.

Il nuovo 1&1 Cloud Server ti offre la combinazione perfetta tra la potenza di un hardware dedicato e la flessibilità del cloud!

FLESSIBILE E CONVENIENTE

Configurazione individuale

- SSD, RAM e CPU sono configurabili in modo flessibile e indipendente e possono essere precisamente adeguate alle tue esigenze

Costi trasparenti

- **NOVITÀ:** tariffazione a minuti
- **NOVITÀ:** panoramica dei costi ben strutturata per una pianificazione e un controllo efficiente delle spese

FACILE E SICURO

1&1 Cloud Panel

- **NOVITÀ:** l'innovativa interfaccia user-friendly con amministrazione intelligente semplifica la gestione del tuo server

Sicurezza

- I data center ad elevate prestazioni di 1&1 sono tra i più sicuri in Europa
- Backup e snapshot permettono di evitare le perdite di dati impreviste
- Il firewall integrato protegge il tuo server da eventuali attacchi

TUTTO INCLUSO

Massime performance

- **NOVITÀ:** attivazione del tuo server in meno di 1 minuto
- **NOVITÀ:** Premium SSD con performance virtuali illimitate
- **NOVITÀ:** reti private, API professionale, load balancing, firewall e molte altre funzioni facilmente configurabili per il tuo server
- **NOVITÀ:** virtualizzazione basata sulla tecnologia leader di VMware
- **NOVITÀ:** applicazioni incluse pronte all'uso, tra cui WordPress, Drupal e Magento
- Parallels® Plesk 12
- Traffico illimitato







DOMINIO | MAIL | HOSTING | SERVER

Inserisci il tuo indirizzo e-mail e inizia subito



PROVALO GRATIS PER 1 MESE!*



1and1.it

800 977313
(numero verde)

* Un mese di prova gratis senza dover indicare i dati di pagamento. In seguito a partire da 15,84 €/mese IVA escl. (configurazione di base). Nessun costo di attivazione. Per maggiori informazioni visita il sito 1and1.it. 1&1 Internet AG, sede in Elgendorfer Str. 57, 56410 Montabaur, Germania.



Michael Shiloh

Smanettone, insegnante, artista, creatore

Michael Shiloh è un artista e insegnante di Arduino che ama smontare le cose per costruire nuove creazioni. Abbiamo parlato con lui dei suoi hobby

È da una vita che Michael Shiloh smanetta con i computer, sostiene i software e i dispositivi gratuiti e Open Source e lavora come istruttore e artista. È attivamente coinvolto nel movimento DIY Maker e i suoi robot e le sue opere d'arte mirano a stimolare le persone a dare vita a creazioni tutte loro. Per questo Michael si serve spesso di schede Arduino.

Linux Pro: Prima di tutto, Michael, il tuo passato è decisamente affascinante: il tuo lavoro spazia dalle creazioni fai-da-te all'elettronica, passando per l'arte. Com'è possibile abbinare Open Source, elettronica e arte?
Michael Shiloh: Ottima domanda. È qualcosa a cui penso molto, tra l'altro, dato che sono anche insegnante e ritengo che la didattica e l'Open Source vadano a braccetto. In un certo senso, l'Open Source fornisce il software ma insegna anche come funziona il software stesso. Ed è ovvio, perché chi è disposto a condividere il proprio software come Open Source vuole anche mostrare al mondo intero come lo ha realizzato. C'è quindi una disponibilità a condividere non solo il codice ma anche il proprio sapere. Credo che in una certa misura ci sia un legame anche tra questo e il mondo dell'arte. Stando alla mia esperienza nel mondo dell'arte, ci sono... beh, stavo per dire che ci sono due tipi di persone ma si tratta di una generalizzazione. Comunque, se vogliamo usare questa espressione, ci sono persone che amano nascondere i propri lavori, in modo tale che l'unica cosa che risulta visibile è l'esterno di ciò che decidono di presentare, mentre il funzionamento interno, cioè il modo in cui ci sono arrivati, rimane nascosto. Altri, invece, sono felicissimi di mostrare a tutti anche l'interno. Questo non vale solo per le arti meccaniche, quelle a cui in un certo senso sono legato io, ma anche alle arti visive, in cui un artista può per esempio utilizzare una tecnica particolare. Può trattarsi magari di un particolare uso del colore e un artista può decidere di dire: "Non vi dirò come ho ottenuto quel colore, è un mio segreto". Un altro, invece, potrebbe dire: "Sono felice di condividere con voi il modo in cui ci sono arrivato: ecco la mia ricetta!". Io penso che il rapporto tra il movimento Open Source e i fautori del 'closed source' sia molto simile. E noto qualcosa di simile anche tra gli spettatori. Quando osservo un'opera d'arte, ho subito voglia di guardarci 'dentro'. Com'è fatta, com'è stata creata? Allo stesso modo, ci sono altri spettatori ai quali basta godersi lo spettacolo così come viene presentato e che non sono curiosi di scoprire come funziona. Come ho già detto, dividere nettamente il mondo in due sarebbe sbagliato; ma a volte si può inquadrare la situazione in questo modo.

LXP: Senz'altro. Come dicevi, ci sono artisti a cui piace avvolgersi nel mistero, separarsi quasi dalla propria arte e presentare le proprie opere senza alcun contesto, mentre altri amano invitare gli altri 'dietro le quinte'.
MS: Sì; e dire questo non significa giudicare gli uni o gli altri. Penso soltanto che ci siano »

due approcci diversi, nessuno dei quali è migliore o più giusto.

LXP: In che ambito rientrano i progetti di cui ti sei occupato?

MS: Direi che ricadono in due categorie, forse tre. Una è quella delle arti meccaniche, a volte chiamate anche arti cinetiche. Sono entrato in contatto con questo ambito grazie al mio lavoro con un gruppo che si chiama Survival Research Labs. Il gruppo è in attività ormai da oltre trent'anni. Realizza delle performance ma a 'esibirsi' sono delle macchine. I membri del gruppo costruiscono grosse macchine, perlopiù comandate a distanza, per poi usarle per mettere in scena degli spettacoli. Gli attori sono le macchine stesse; gli esseri umani fanno da operatori, dietro le quinte o fuori dal palco. Molte delle mie creazioni di arte meccanica si ispirano fortemente a questo tipo di lavori. Inoltre ho realizzato molti oggetti di arredamento che in un certo senso sono simili a questi lavori, dato che uso molto il metallo e il cemento. Insomma, uso dei materiali industriali. Materiali industriali pesanti, che a mio modo di vedere hanno un legame con le arti meccaniche. Di solito questi lavori sono di natura concettuale. Mi interessa molto l'aspetto concettuale dell'arte. La terza categoria è qualcosa che ho grande difficoltà a definire. Si tratta di creare situazioni in cui le persone devono fare qualcosa perché l'opera d'arte acquisti un significato. Un esempio è un robot che ho costruito molto tempo fa, una ventina d'anni fa. Ho dovuto scavare negli archivi di Internet per scovare il sito Web originale. Era un robot molto semplice, con pochi sensori e un paio di motori: e il computer che lo controllava era un server Linux completo. Era su Internet, accessibile a tutti. Era collegato a una pagina Web e c'era un dominio correlato, **http://linuxrobot.org**. Praticamente, il robot non faceva nulla finché qualcuno non si collegava al sito e scriveva un programma per farlo agire.

Sulla pagina Web fornivo alcuni esempi e spiegavo come funzionava il tutto. Era tutto documentato, c'erano gli esempi e le istruzioni per effettuare il collegamento. Mettevo a disposizione la password di root: l'idea era che persone di tutto il mondo si collegassero, scrivessero programmi e facessero fare qualcosa al robot. È stato un esperimento molto interessante: per la maggior parte della giornata il robot non faceva nulla e poi all'improvviso si svegliava e cominciava a muoversi! Di solito capitava in piena notte, il che suggerisce che la maggior parte dei miei visitatori abitasse dall'altra parte della Terra. O forse stavano semplicemente svegli in piena notte. Molti non lasciavano alcun commento: non avevo idea di chi fossero, che cosa facessero o da dove venissero. Uno dei miei preferiti era qualcuno che voleva far ballare il robot. Faceva esperimenti con vari movimenti, in modo da farlo ballare. A me non sarebbe mai venuto in mente. Insomma, l'idea di mettere a disposizione una specie di lavagna vuota e chiedere quindi alla gente di riempirla con qualcosa mi intrigava; e in seguito ho realizzato diversi progetti simili. Sto ancora riflettendo su come potrei fare... A volte ho delle difficoltà a presentare questi lavori come opere d'arte.

Ho creato varie opere dello stesso tipo. Abbiamo realizzato un progetto analogo con un altro robot, una versione più piccola; e lo abbiamo presentato in una galleria d'arte. Le persone dovevano sedersi e programmare sul posto. Incredibile a dirsi, lo facevano davvero. L'ho sempre detto: se un'opera esposta in una galleria d'arte riesce a competere con gli stuzzichini e il vino gratuiti, significa che ha veramente avuto successo...



OPERE D'ARTE ROBOTICHE

"Il gruppo Survival Research Labs organizza performance ma a esibirsi sono le macchine..."

LXP: Fantastico. Sarebbe interessante riflettere su quanti tra i visitatori di una normale galleria d'arte sarebbero in grado di mettersi a programmare. Sono queste persone il tuo pubblico di riferimento?

MS: Non necessariamente. Quella era una galleria d'arte normale e abbiamo realizzato il progetto con Arduino, perciò era molto semplice; inoltre fornivamo esempi e spiegavamo alle persone che cosa stava accadendo. Comunque, hai perfettamente ragione: non è roba per tutti. Però c'erano anche persone che non avevano mai programmato nulla in vita loro e che si sono sedute a programmare. Un mio amico ha detto che il trucco è non dire loro che stanno programmando. Dite loro solo che stanno modificando qualcosa. Date loro qualche dritta riguardo a cosa modificare e cosa attendersi quando si modifica qualcosa in un certo modo, dopo di che lasciateli smanettare. In questo modo non si renderanno nemmeno conto del fatto che stanno programmando. Questo in realtà è stato un altro aspetto interessante del progetto. Mi interessa molto il risvolto della didattica tecnologica per cui, se si dice a una persona che dovrà misurarsi con qualcosa di complicato come la programmazione e questa persona ha una sorta di blocco o non si considera un programmatore, non avrà il coraggio di farsi avanti. Se invece si riesce in qualche modo a non presentare l'attività come tale, allora si faranno avanti più persone che diversamente avrebbero lasciato perdere.

Io l'ho imparato in occasione del primo Maker Fair, quando abbiamo realizzato un enorme laboratorio il cui scopo era mettere le persone in grado di smontare le cose e costruire qualcosa di nuovo. La cosa ha funzionato perché c'era un sacco di spazio. Intere famiglie arrivavano e si sedevano; di solito dicevano "Andiamo solo a dare un'occhiata" e un solo membro della famiglia veniva designato a partecipare attivamente. Di solito era un ragazzino maschio, come vuole lo stereotipo. Dopo un po', tuttavia, anche i genitori si mettevano a trafficare con gli oggetti. Ce n'erano dappertutto, perciò il materiale non mancava. Gli altri bambini cominciavano a loro volta a giocare con le cose, magari cominciando con qualche oggetto vicino a loro. Guardavano che cosa facevano le altre persone e cominciavano ad aggiungerci qualcosa. Dopo qualche ora i partecipanti cominciavano a costruire splendidi ed enormi aggeggi; e la cosa più interessante per me era che nessuno di loro si sarebbe fatto avanti se avessi detto loro "Qui

costruirete qualcosa". Invece, semplicemente in virtù del fatto che c'era spazio per sedersi, partecipavano. È questo aspetto, in un certo senso, a spingermi a realizzare i progetti con i robot. Come posso indurre le persone a misurarsi con questi lavori senza che sollevino delle barriere, senza che dicano "Oh, non sono capace"?

LXP: Il termine 'programmazione' ha delle connotazioni che spingono molte persone a dire: "Beh non sono laureato in informatica, perciò non ne sono capace". E se invece le si invitasse a far ballare un robot? Questo introduce un che di umano, ed ecco che all'improvviso la gente dice: "Magari posso riuscirci!"..

MS: Hai ragione; e penso che questo sia interessante sul piano didattico, proprio per questo motivo. Il punto, per quanto riguarda l'insegnamento della robotica, dell'elettronica e della programmazione, è che moltissimi studenti graviteranno verso questo tipo di materie, in un modo o nell'altro. Poi ci sarà un altro gruppo di studenti che le abbandonerà comunque, qualunque cosa si faccia. È questo secondo gruppo che mi interessa. Capisci, il primo gruppo ce la farà comunque; ma come possiamo raggiungere il secondo?

LXP: La scoperta dell'Open Source per te è avvenuta in modo spontaneo nel corso della tua carriera? Ha rappresentato uno sviluppo della tua attività artistica?

MS: Posso dire di aver scoperto l'Open Source molto tempo prima che esistesse una parola per definirlo. Ho frequentato l'università a Berkeley nel periodo in cui Unix era in via di sviluppo e Internet (anche se allora non si chiamava così) stava appena iniziando a nascere. La gente condivideva le cose in modo estremamente libero, perché allora sembrava che non avessero alcun valore in sé! Perciò, se qualcuno scriveva un programmino interessante... Per esempio, una mia collega voleva sapere quando riceveva un nuovo messaggio di posta elettronica e ha scritto un programmino che faceva suonare un campanello ogni volta che arrivava un nuovo messaggio. Oggi naturalmente qualunque programma di posta elettronica ha una funzione del genere ma allora non era così.
Era straordinariamente utile e si è diffuso tramite Internet; non penso che la mia collega lo abbia distribuito ufficialmente come open source ma era assolutamente scontato che le cose andassero così. Credo sia stata l'abitudine culturale a condividere le informazioni in questo modo, più che il closed source, a portarmi verso l'Open Source.
È stato solo quando ho cominciato a imbattermi in programmi che non erano fatti in quel modo che ho pensato: "Ehi, esiste tutto un altro mondo!"

LXP: La tua formazione, quindi, è molto legata allo smanettare e alla realizzazione rapida di prototipi: è stato questo a portarti ad Arduino?

MS: Vedi, non saprei dire esattamente che cosa mi abbia portato ad Arduino ma probabilmente è stato qualcosa del genere. Sono sempre stato uno smanettone, non è che un giorno mi sia svegliato e abbia deciso di diventare tale!
È semplicemente il mio modo di lavorare: trafficare con componenti elettronici di ogni tipo, costruire cose, inventare. Non ricordo il momento esatto in cui mi sono imbattuto in Arduino. Probabilmente ho semplicemente cominciato a sentire questo nome e ho deciso che volevo saperne di più. A dire il vero, molti anni fa insieme ad altri ho fondato un'azienda per fare ciò che oggi fa Arduino; e non eravamo i soli. C'erano altre aziende che si occupavano di cose simili. Per esempio all'epoca c'era BASIC Stamp. Era un periodo in cui molti computer avevano ancora porte seriali. Penso che il nuovo computer Apple sia stato il primo a non avere una porta seriale. Tutti quanti erano nel panico: che cosa avremmo fatto con tutte le schede che utilizzavano le porte seriali? Così, con un paio di amici ho fondato un'azienda per realizzare un progetto sostanzialmente simile ad Arduino. Aveva una connessione USB; usava anzi lo stesso chip che in seguito è stato utilizzato per Arduino, il chip FTDI. Disponeva inoltre di un certo numero di porte I/O generiche che potevano essere configurate in vari modi. Abbiamo avuto un discreto successo: siamo entrati in un paio di centinaia di università sparse in tutto il mondo, se non sbaglio anche l'Università di Bath aveva uno dei nostri dispositivi. Tuttavia il successo non è stato travolgente. Il progetto non è esploso come poi è accaduto ad Arduino. Dopo aver seguito il progetto per alcuni anni, più o meno nel periodo in cui è entrato in circolazione Arduino, abbiamo ritenuto che non fosse più necessario. Io tenevo comunque gli occhi aperti per scovare qualcosa di simile ed è stato così che ho sentito parlare di Arduino. I diversi tipi di schede si differenziavano tra loro per funzioni, prezzo, livello di complessità e approccio al software e penso che uno dei motivi del successo di Arduino sia stata la sua capacità di offrire una soluzione equilibrata in tutti questi ambiti: quanto doveva essere complesso, che cosa doveva fare il software, che cosa doveva fare l'utente, quale doveva essere il prezzo. Un'altra magia di Arduino è stata la nascita immediata di una grossa comunità, che credo sia stata decisiva per il suo successo. Questa comunità, infatti, è stata utile non solo sotto l'aspetto della convalida ma anche nel fornire l'assistenza che una piccola azienda non è semplicemente in grado di offrire.

STORIA DEL COMPUTER

"Ho scoperto l'Open Source molto prima che esistesse la parola per definirlo..."

LXP: Questo dimostra anche la potenza dei progetti aperti e la loro capacità di crescere rapidamente. Oggi l'ecosistema è enorme e attira una comunità estremamente attiva. Perciò, l'hardware aperto appare come uno sviluppo logico del software Open Source; eppure i progetti in questo campo sono ancora relativamente poco numerosi. Quali sono gli ostacoli a tuo parere?

MS: Direi che gli ostacoli principali sono probabilmente due. Uno sta nel fatto che mentre il software Open Source può essere riprodotto semplicemente copiando un file, nel caso dell'hardware aperto occorre produrre fisicamente qualcosa. Credo che un'altra difficoltà sia il fatto che a mio parere il Software Libero, come attività, dà la sensazione che sia possibile guadagnare denaro in altri modi: per esempio, offrendo assistenza o versioni personalizzate del software stesso. Con l'hardware, invece, l'idea è che l'unico »



modo per guadagnarci qualcosa consista nel venderlo e che quindi una volta che si dà alla gente la possibilità di produrselo da sé, esso risulti inutile per chi lo ha realizzato.

LXP: Ritieni che l'hardware aperto continuerà comunque a crescere malgrado questi ostacoli?
MS: Buona domanda. Io spero di sì e penso che probabilmente sarà così; di certo, al momento c'è un grosso movimento a favore dell'hardware aperto. Tuttavia, alla luce di questi ostacoli, forse... il motivo della mia esitazione è questo. Attualmente c'è grande interesse per il movimento dei maker e del fai-da-te. E ovviamente, il fai-da-te non è un concetto nuovo: si può dire che è antico quanto la nostra specie. Ciò a cui stiamo assistendo oggi, direi, è una ripresa dell'interesse. Penso che sia già accaduto ciclicamente. Non sono uno storico ma ho la sensazione che quando la produzione di massa ha iniziato a prendere il sopravvento, sia stata necessariamente accompagnata anche da un cambiamento di mentalità: il consumo di massa, naturalmente. Perché il consumo di massa funzioni, tuttavia, occorre che la gente sia disposta a buttare via le cose per comprarne di nuove. Una persona ha bisogno di un solo telefono; ma se quel telefono diventa obsoleto nel giro di sei mesi, deve comprarsene un altro. Penso che questo abbia scoraggiato gli amanti del fai-da-te, gli smanettoni e i creatori, perché quando le cose vengono prodotte in massa è necessario produrle nel modo più economico possibile, il che significa che devono essere quanto più compresse possibile. E questo le rende più difficili da modificare. Oggi c'è una rinascita dell'interesse per il movimento del fai-da-te ma la mia paura è che si tratti solo di un altro di questi cicli, che l'interesse aumenti per poi diminuire di nuovo. Credo che l'interesse per l'hardware aperto sia legato a questo, perché buona parte dei prodotti di questo settore si rivolge a questa comunità. Se questa comunità diventerà meno importante, temo che anche l'hardware aperto farà la stessa fine. La mia speranza è che questo non sia semplicemente un ciclo ma qualcosa destinato a continuare, una tendenza in ascesa. Se l'interesse per il fai-da-te è davvero una tendenza in ascesa, allora sì, credo che anche il movimento per l'hardware aperto continuerà a rappresentare una tendenza in ascesa.

LXP: Di sicuro, il software Open Source ha anche un risvolto etico, quello di mettere il software a disposizione di persone che non sarebbero in grado di acquistare una costosa licenza per il prodotto proprietario equivalente. Con l'hardware aperto, come dicevi, a prescindere dal desiderio di offrire alla gente materiali e strumenti per costruirsi i dispositivi, questo non è sempre possibile. Nel tuo profilo LinkedIn affermi che gli approcci etici alle risorse sono molto importanti per te: in che modo ti assicuri che questi approcci siano sempre presenti in ciò che fai?
MS: Non so dire se ci riesco. Mi è sempre interessato riparare le cose e senz'altro c'è un forte risvolto etico in questo. Inoltre c'è una sorta di risvolto estetico, non so come dire... mi fa sentire bene. Probabilmente anche qui c'entra l'etica. Mi sento bene quando riesco a riparare o a riutilizzare un oggetto invece di comprarne uno nuovo. Direi quindi che nel mio lavoro non ho bisogno di sforzarmi di farlo, perché mi viene naturale, è una mia tendenza. Quando insegno queste cose o quando preparo un laboratorio, in molti dei miei corsi utilizzo componenti elettronici di scarto o riciclati. Non sono il massimo per un'esercitazione da tre ore; però ho appena terminato un laboratorio della durata di due settimane per la California State University. Dato che avevamo a disposizione due settimane, era possibile farlo: così, prima di iniziare il laboratorio abbiamo fatto raccogliere agli studenti un certo numero di componenti elettronici di scarto, abbiamo parlato loro di come smontarli e di quali elementi utili avremmo trovato al loro interno, in modo da insegnare loro a distinguere i vari componenti e a riutilizzarli. Dopo di che, li hanno usati per costruire cose nuove. Penso quindi che dimostrando che questo è possibile e mettendo a loro disposizione questi materiali siamo riusciti a trasmettere loro fin da subito questo tipo di etica.



RASPBERRY PI

"Non sono un fanatico dell'Open Source ma preferirei che il Pi fosse aperto"

LXP: Arduino è un progetto piuttosto aperto; è interessante il contrasto con il Raspberry Pi, perché una delle principali lamentele è che malgrado la sua popolarità il Pi contiene comunque elementi closed source. Che cosa ne pensi?
MS: Non ho un'opinione precisa a riguardo. Non sono un fanatico dell'Open Source: uso un Android invece di un iPhone ma d'altronde il telefono in sé contiene elementi closed source. Preferirei che il Raspberry Pi fosse aperto, perché ritengo rappresenti un'opportunità didattica che al momento va perduta. Non ne faccio una questione di 'questo contro quest'altro': è semplicemente un'ulteriore possibilità all'interno di questo oceano di schede in grado di svolgere questo tipo di funzioni.



LXP: Certo; ciò che conta è il risultato finale, che è stimolare i bambini a creare progetti tutti loro, entro limiti di spesa sostenibili. Ritieni che questo sia stato considerato più importante delle preoccupazioni relative a quanti fossero gli elementi closed source?

MS: Senz'altro. L'obiettivo di Eben era sostanzialmente fare sì che questi computer finissero nelle mani di quante più persone possibile; e se è riuscito a ridurre ulteriormente il prezzo accettando di includere nel Pi elementi closed source, direi che ha raggiunto il suo obiettivo.

LXP: È vero; e noi di Linux Pro siamo decisamente entusiasti di come le scuole britanniche stanno adottando il Raspberry Pi. Ritieni che Arduino stia svolgendo un ruolo analogo negli USA e nel resto del mondo?

MS: Sì, sia negli USA sia nel resto del mondo, in particolare in Europa. Lo vedo in un numero sempre maggiore di scuole e università.
Mi incuriosisce sempre scoprire come lo abbiano scoperto e come abbiano deciso di usarlo. Non mi sorprende che venga adottato dalle università ma quando lo vedo apparire nelle scuole superiori sono sempre piacevolmente stupito. Immagino che nella maggior parte dei casi ci sia un insegnante che lo conosce e pensa: "Questo sarebbe utile per ciò che facciamo". Ciò che apprezzo è che non viene usato solo per insegnare programmazione o robotica o per gli impieghi per cui è stato progettato: gli insegnanti cominciano a considerarlo uno strumento utile per parlare di qualunque cosa decidano di parlare.
Un esempio sono le scienze ambientali, come il monitoraggio dell'atmosfera, dell'acqua e simili; e ho parlato con un insegnante di fisica che intendeva usare Arduino per monitorare una macchinina che scendeva lungo un piano inclinato. Insomma, tutti i classici esperimenti di fisica, realizzati con l'aiuto di Arduino. Allora gli ho chiesto: "Arduino sta contribuendo davvero alla tua attività didattica?" e lui ha risposto che la cosa più importante è che i bambini ci si appassionano veramente. Sono davvero interessati a questa piccola scheda con la lucina intermittente. Perciò, mi ha detto, se Arduino li fa interessare di più alle cose che cerco di insegnare loro, allora voglio usarlo! Insomma, è diventato una sorta di veicolo per stimolare la classe e questo è grandioso.

COME SI FA
"Non ci facciamo pubblicità ma la gente continua a trovarci e a chiederci 'come si fa'"

LXP: E per quanto riguarda Arduino Yún, l'interessante ibrido Linux-Arduino? Dove ritieni che potrebbe essere utilizzato in particolare?

MS: Sono al corrente di numerosi progetti individuali, che tuttavia non mi pare rientrino in categorie specifiche. L'ambito in cui vedo usare più comunemente lo Yún sono tutti quei progetti che richiedono la connessione a Internet: disponendo di funzioni Ethernet e Wi-Fi incorporate, è ideale per questo tipo di impieghi. Anzi, il laboratorio che ho appena realizzato per OSCON era dedicato proprio a questo tema. Se l'obiettivo è controllare un robot in network, monitorare un'area o utilizzare applicazioni di controllo o di rilevazione di qualunque genere che debbano risultare accessibili tramite rete... lo Yún è l'ideale. Il pubblico di riferimento originale di Arduino non erano i tecnici. Si trattava cioè di capire come stimolare i non-tecnici ad accendere e spegnere dei LED o a riconoscere le pressioni di determinati tasti; di mostrare loro come farlo senza dover diventare dei tecnici. Poi è uscito lo Yún: adesso comprende un computer vero e proprio, il livello tecnico è superiore, perciò la domanda che devo pormi quando realizzo dei laboratori è: come integrare tutto questo in quella filosofia?
Gli sviluppatori dei software di Arduino hanno creato un meccanismo per cui dall'interno di Arduino è possibile innescare degli eventi sul piano di Linux; per esempio, è possibile avviare un programma in Linux, eseguire un comando e ottenere una reazione. È possibile ordinare a Linux di andare su Internet, scaricare dei dati e restituirli.

LXP: In quale direzione vedi quindi muoversi il futuro di Arduino?

MS: Premesso che questo non è il punto di vista ufficiale di Arduino ma solo una mia interpretazione di ciò che vedo e sento, ovviamente escono di continuo nuovi processori e nuovi micro-controller, perciò penso che 'arduinizzare' quelli più adatti sarebbe uno sviluppo interessante. Mano a mano che questi dispositivi diventeranno più potenti, la gente inizierà a mettere da parte i modelli più antiquati, perciò dovremo rimanere al passo con i tempi ove necessario. Penso che Arduino Yún abbia dimostrato l'importanza di un utilizzo semplice della rete, perciò immagino che in futuro assisteremo ad altri sviluppi e progressi in questo senso.
La didattica è un ambito decisamente importante per noi, perciò sarà decisivo scoprire nuovi modi di utilizzarlo nella scuola, come dicevamo poco fa. Si tratta di capire come continuare in questa direzione e come rendere Arduino più accessibile, di mettere a disposizione ulteriori materiali per questi fini. Queste sono le tendenze di cui sono al corrente; posso ipotizzare molte cose ma non ho nessuna certezza...

LXP: Per finire, hai partecipato alla fondazione di Teach Me To Make. Ha avuto successo?

MS: Molto più di quanto sperassimo. In origine avevo creato Teach Me To Make perché per il laboratorio Maker Fair di cui parlavo prima mi ero accorto di avere bisogno di un sito Web e di un URL in cui collocarlo. Così mi è venuto in mente quasi per caso Teach Me To Make e mi sono accaparrato il dominio giusto per potervi archiviare materiali, immagini e via dicendo. Quanto a Teach Me To Make come organizzazione, è nata quando la mia collaboratrice Judy e io abbiamo iniziato a realizzare laboratori insieme: ci serviva un posto in cui tenere i nostri materiali. Avevamo bisogno di un posto in cui pubblicizzare le nostre attività, di un sito sul quale fornire informazioni sui nostri nuovi laboratori, sugli eventi a cui partecipavamo e sui nuovi progetti che realizzavamo. Il sito comprende un blog Wordpress: è facile da usare e mi piace avere la possibilità di modificare istantaneamente i contenuti. Wordpress è ideale per questo. Ho finito per pubblicare una quantità di materiale, di solito ispirato dalle domande che mi pongono i miei studenti. Una delle domande ricorrenti era: "Come si fa a usare un transistor come interruttore?". Così, ho trattato l'argomento in modo piuttosto esauriente. Incredibile a dirsi, il sito riceve tuttora visite legate a questo tema. Ricevo domande da tutto il mondo riguardo a come fare determinate cose. Insomma, il sito è frequentato; anche se non pubblicizziamo molto Teach Me To Make, la gente continua a trovarci e a chiederci 'come si fa'. LXP

Accendi, sintonizza, usa JACK!

Vi guidiamo alla scoperta di JACK e con l'aiuto di libremusicproduction.com entrerete nel mondo dell'audio professionale con Linux

JACK è il miglior server Linux dedicato alla lavorazione professionale dell'audio. Il suo nome deriva dall'acronimo di *JACK Audio Connection Kit*. Prima di proseguire è però importante fare una premessa: l'approccio a JACK non è dei più semplici e, se siete nuovi nell'uso di questi strumenti, potreste percepire un certo disorientamento iniziale. Questo, però, non deve intimorirvi più di tanto. Infatti è sufficiente superare la scorza superficiale per prendere confidenza con le funzioni più utili. Dovete solo imparare a conoscere le basi del programma, dopodiché sarete in grado di lavorare senza alcun problema. Alla fine di questa guida ci auguriamo che abbiate acquisito una sicurezza tale da continuare a usare JACK con naturalezza.

Che cos'è JACK

JACK è un server audio ottimizzato per la gestione e la produzione di brani musicali. Facciamo quindi una rapida panoramica delle sue funzioni principali.

Impostazioni: JACK controlla le impostazioni audio e MIDI. Consente quindi di scegliere l'interfaccia audio più adatta, nonché tutte le impostazioni derivate come la frequenza di campionamento, la dimensione del buffer e molto altro ancora. Tratteremo poi queste opzioni più avanti nell'articolo.

Performance: Usando JACK potrete sfruttare latenze molto basse sia con audio standard sia con i MIDI. In altre parole, se state registrando uno strumento sul PC, mentre lo suonate, potrete controllare il ritorno su cuffie o altoparlanti senza alcun ritardo. Per approfondire i concetti di latenza e compensazione, date un'occhiata all'indirizzo **bit.ly/LXFardour**.

Collegamenti e interconnettività: Questi sono i punti forti del server. Qualsiasi ingresso o uscita della vostra interfaccia audio o di un programma compatibile può essere connesso arbitrariamente insieme. Questo significa che avrete a disposizione gli stessi standard di collegamento per tutte le applicazioni pensate per funzionare con JACK. Se ci pensate bene, si tratta di un'interoperabilità enorme, che vi consente di sfruttare uno strumento altamente flessibile e capace di connettere una vasta varietà di programmi.

Tutto sincronizzato

JACK può essere utilizzato per sincronizzare tra loro un'enorme quantità di software. Se per esempio volete utilizzare il sequencer MIDI di un programma con il sequencer audio di un'altra applicazione, niente vi impedisce di farlo. Entrambi i software riusciranno a lavorare perfettamente tra loro con i medesimi tempi. Il gesto di muoversi nella timeline di un programma si rifletterà nel medesimo movimento su tutte le altre applicazioni sincronizzate. Come abbiamo già avuto modo di anticipare, JACK, pur essendo uno strumento piuttosto complicato, non deve intimidirvi oltremodo. Anche se il server riesce a gestire flussi di lavoro molto complessi, non significa che sia inadatto a essere sfruttato tramite impostazioni basilari. In pratica, vi permetterà di fare qualsiasi cosa abbiate in mente: dalle composizioni più semplici fino a quelle più complesse. Adesso che sapete di cosa è capace JACK, il passo successivo è capire come utilizzarlo. Prima di tutto va detto che il server è di per sé uno strumento a riga di comando. Tuttavia esistono diversi gestori grafici che consentiranno di rendere l'approccio a JACK più semplice del previsto. Su questo fronte ci sono due tipi di applicazioni principali che possiamo inserire in altrettanti macro insiemi: i *JACK set-up manager* e i *JACK connection manager*. I primi permettono di avviare il server tramite una serie di impostazioni predefinite, i secondi, invece, si occupano principalmente di gestire e creare i collegamenti. I vari gestori grafici di JACK riescono comunque a combinare questi due aspetti a vari livelli. Per scegliere adeguatamente è quindi necessario valutare alcuni fattori. Se state lavorando all'interno di una *Digital Audio Workstation* (DAW) e utilizzate programmi esterni, potrebbe essere più indicato un JACK set-up manager che vi permette di avviare il server con la configurazione desiderata. In questo caso, potrete regolare tutte le connessioni software all'interno della DAW, così che l'uso di un JACK connection manager risulti del tutto inutile. Per chi invece incentra il proprio lavoro su una registrazione modulare, con processori multipli di effetti, software di registrazione e sequenzionamento, un gestore JACK più improntato alle connessioni potrebbe essere la scelta migliore. Questo anche in virtù del fatto che molti programmi modulari non possiedono alcun gestore per i collegamenti. Di seguito riportiamo quindi i gestori per JACK tra i più popolari. **Qjackctl** è tra i JACK manager più potenti. Consente di accedere a una grande quantità di impostazioni e in più include un gestore di connessione e un'applicazione per la regolazione delle sessioni. Si presenta sotto forma di una piccola finestra che consente di accedere a tutte le impostazioni più importanti. Può anche essere rimpicciolito nella barra delle applicazioni, in modo da renderlo visibile solo quando ce n'è bisogno. **Cadence** è un altro strumento utile per l'uso e l'avvio di JACK. Lo troviamo come parte integrante della distro **KXStudio** insieme a molti altri componenti del server che permettono di sfruttare a pieno sia i suoni sia i video in flash. Cadence consente »

Alla scoperta di JACK



1 Seleziona un driver
Selezionate la scheda **Driver** di Cadence, quindi scegliete il driver più appropriato. Di solito si tratta di **ALSA**.

2 Scegli l'interfaccia
Assicuratevi che l'interfaccia sia collegata e accesa, altrimenti potrebbe non essere selezionabile.

3 Regola le impostazioni
Abbinate le impostazioni dell'interfaccia qui. A tal proposito, tenete presente quanto detto nell'articolo.

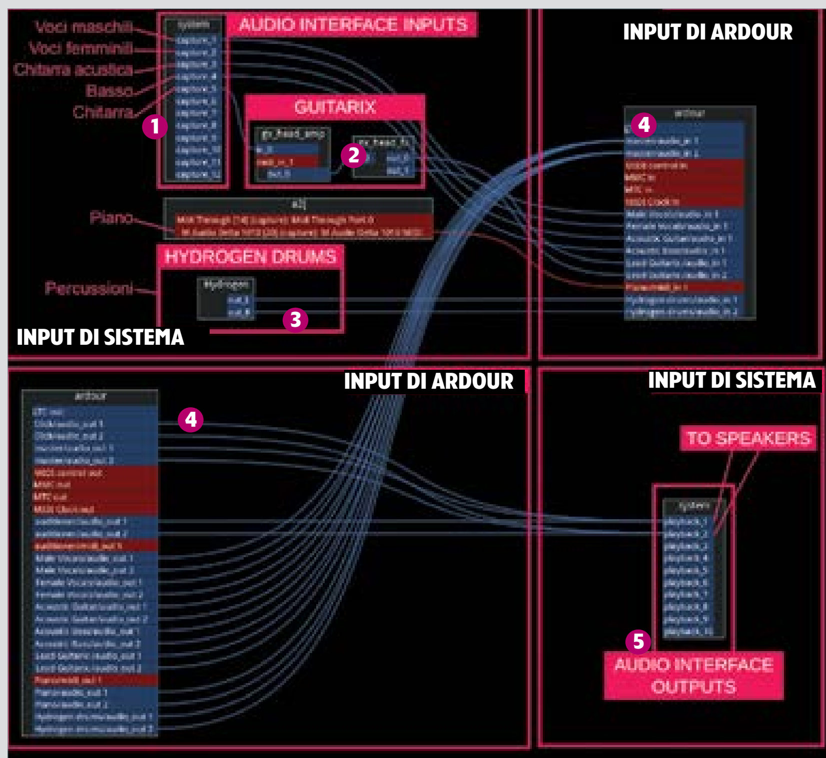
Compatibilità con JACK

La maggior parte dei software audio supporta senza problemi JACK. Questo significa che gli ingressi audio e MIDI sono gestiti attraverso il server. Se siete abituati a gestire tutte le connessioni in una DAW (*Digital Audio Workstation*), pensate a JACK allo stesso modo. L'unica differenza è a livello di sistema. Qualsiasi software compatibile con JACK può essere inserito all'interno di questo insieme, in modo assolutamente simile a come un plug-in viene implementato in una DAW. La differenza principale è che JACK permette di avere un controllo estremamente preciso sui collegamenti.

Monolitico

Quando parliamo di set-up monolitico ci riferiamo a una configurazione dove il lavoro viene gestito all'interno di un unico programma. L'approccio monolitico è il più utilizzato dai sistemi Windows e Mac, mentre Linux si distingue per il suo contrario, ovvero la modularità.

JACK in Patchage



1 Audio interface inputs Gli input principali del brano sono elencati qui.

2 GuitarIX Un amplificatore per chitarra virtuale molto valido da usare.

3 Hydrogen drums Gli input della drum machine GNU/Linux.

4 Ardour Fondamentale per l'editing, la registrazione e la miscelazione.

5 Audio interface outputs Gli input sono miscelati qui e inviati alle uscite.

anche di avviare JACK tramite log-in. **Patchage** e **Catia**, infine, sono due JACK connection manager dotati di interfaccia grafica. Offrono una panoramica precisa e puntale di tutti i collegamenti, con in più l'uso di una serie di colori per le porte principali audio e MIDI. In questo modo, potrete identificarle ancora meglio e rendere il processo d'uso molto più semplice.

Impostazioni di base

Se siete già abbastanza esperti nell'uso di altre piattaforme per la gestione e la creazione audio, sicuramente non avrete alcuna difficoltà a familiarizzare con le principali impostazioni di JACK. Se invece siete dei neofiti, inizialmente potreste sentirvi sopraffatti dall'enorme quantità di opzioni presenti nel nostro server. Il primo passo per procedere, in questo caso, è non farsi distrarre e ricordare che la maggior parte delle funzioni non verranno utilizzate in questa fase iniziale. Di seguito illustriamo le principali voci di cui tenere conto:

Buffer size (frames/period): buffer di dimensioni minori producono meno latenza. Un valore più basso, però, metterà alla prova il PC che dovrà essere piuttosto veloce. Infatti, ottenere una latenza ridotta va a discapito della CPU, che verrà utilizzata in modo più corposo. Buffer più ampi sono di norma consigliabili, anche se in questo caso non potrete ottenere una bassa latenza. Se state cercando di realizzare una creazione con latenza ridotta, cercate di impostare il buffer tra 64 e 256.

Sample Rate: alte frequenze di campionamento producono una minor latenza con le stesse impostazioni del buffer. Questa impostazione dipende dalla configurazione ottimale dell'interfaccia audio, nonché dalle vostre preferenze. Alcuni preferiscono registrare con frequenze di campionamento superiori, altri, invece, si accontentano della qualità CD pari a 44,100 kHz (44,1). Uno dei valori più utilizzati è 48,000 kHz che permette di mettere d'accordo un po' tutti. Da tenere presente che più aumentiamo la frequenza di campionamento, più bassa sarà la latenza. Questo, però, metterà a dura prova la CPU che dovrà gestire un carico di lavoro piuttosto oneroso.

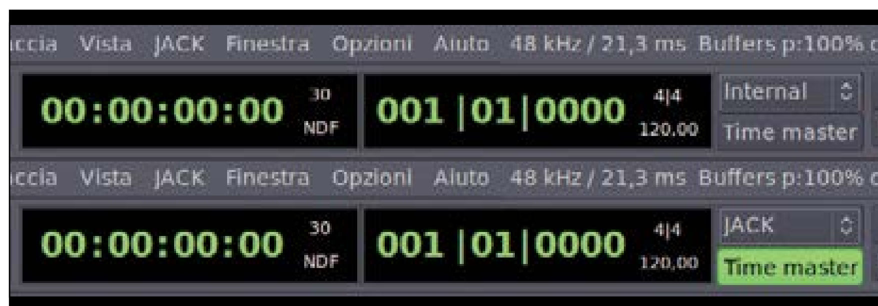
Periods/buffer: se state utilizzando un dispositivo USB, potete ottenere una bassa latenza regolando questa impostazione con il valore 3. In caso contrario, utilizzate la funzione 2. Se volete avere a tutti i costi una latenza ridotta, vi consigliamo di approfondire il funzionamento di questa opzione, così da evitare di mandare in burn-out la CPU. Se non avete bisogno di monitorare l'audio di ritorno dal PC a bassa latenza, probabilmente non ci sarà alcuna necessità di spingere il processore oltre i livelli di guardia. In questo caso è sempre meglio utilizzare una latenza un po' più alta ed evitare il sovraccaricamento del sistema.

Configurare JACK

KXStudio (**http://kxstudio.sourceforge.net**) e AV Linux (**http://www.bandshed.net/AVLinux.html**) sono due distribuzioni Linux che forniscono tutto quanto avete bisogno per seguire questa guida. Vedremo quindi come configurare le impostazioni di JACK sia con Qjackctl sia con Cadence. Le operazioni che descriviamo di seguito vanno bene per entrambe le applicazioni. Dove così non fosse, ve lo facciamo notare.

Passo 1: in Qjackctl fate click sul pulsante **Impostazioni** o su **Configure** in Cadence.

Passo 2: Assicuratevi che la funzione **Tempo Reale** sia selezionata (in Cadence è **Realtime** nella scheda **Engine**).



› **Per sincronizzare i programmi, uno deve essere master e gli altri slave**



› In Ardour, la matrice di collegamento è davvero molto semplice da gestire

Passo 3: Selezionate il vostro driver audio (assicuratevi che la scheda **Driver** sia attiva in Cadence). Se usate un dispositivo firewire, selezionate **Firewire**. In tutti gli altri casi, scegliete **ALSA**.
Passo 4: Selezionate la vostra interfaccia dal menu **Interface**.
Passo 5: Scegliete le impostazioni per l'interfaccia.
Passo 6: Fate click su **OK** per applicare le impostazioni.
Passo 7: Premete **Start** su Cadence e **Avvia** su Qjackctl.
In Qjackctl non toccate il campo **Nome**. Alcuni principianti non sanno rinunciare alla possibilità di rinominare il proprio server ma questo può causare alcuni malfunzionamenti. In altre parole, in questa fase, è sempre meglio non mettere mano alle opzioni che non si conoscono.
Una volta regolate le impostazioni necessarie, potrete utilizzare tutti i programmi compatibili con JACK attraverso la stessa configurazione. Alcune applicazioni consentono poi di modificare le dimensioni del buffer, così da regolare la latenza senza riavviare JACK. In definitiva, anche se il nostro server è particolarmente potente, non c'è motivo per sentirsi intimiditi. Come dimostrato, nella sua forma più semplice la configurazione dell'interfaccia è davvero intuitiva. Come con qualsiasi buon programma, naturalmente è possibile andare oltre le normali impostazioni di routine. Se vi spingete avanti e nel farlo doveste riscontrare alcuni problemi dovuti al riconoscimento dell'interfaccia, basterà assicurarsi che questa sia collegata e attiva prima ancora di avviare JACK. Se continuate ad avere problemi, provate a riavviare il sistema mentre l'interfaccia rimane collegata.

JACK Sync

Un altro aspetto nell'uso di JACK utile da conoscere è la sincronizzazione. Non è solo possibile interconnettere programmi, ma si può anche fare in modo che il loro utilizzo sia perfettamente sincronizzato. Ecco come operare.

Passo 1: prima di tutto avviate JACK tramite il vostro JACK manager preferito.
Passo 2: avviate tutti i programmi che volete sincronizzare. In questo caso abbiamo scelto **Ardour** e **Hydrogen**. Naturalmente controllate che entrambi i software siano correttamente configurati.
Passo 3: Per impostazione predefinita, Ardour è impostato sul timemaster di JACK che corrisponde esattamente a ciò che vogliamo. Questa impostazione può essere trovata navigando in **Sessione ➠ Proprietà**, passando poi nella scheda **Timecode**.
Passo 4: Adesso fate click sul pulsante **Internal** nella toolbar di Ardour e selezionate la voce **JACK**.

Passaggi per Hydrogen

Passo 1: Andate in **Strumenti ➠ Preferenze**. Nella scheda **Sistema** assicuratevi che sia selezionato **JACK**, quindi riavviate il driver.
Passo 2: Nella barra degli strumenti noterete due pulsanti timemaster. Assicuratevi quindi che **J.trans** sia abilitato.
Passo 3: Adesso entrambi i programmi sono sincronizzati. Inoltre, avendo Ardour funzioni di timemaster, modificando il tempo in quest'ultimo programma l'impostazione si rifletterà anche in Hydrogen o qualsiasi altro software slave. Il fatto che JACK sia in grado di gestire porte audio e MIDI, oltre a permettere a tutte le applicazioni che lo utilizzano di restare sincronizzate tra loro, rende il server davvero potente. Così facendo, infatti, potete creare applicazioni modulari in modo davvero veloce. Per chiunque sia interessato ad allestire delle configurazioni modulari, vi consigliamo di dare un'occhiata alla gestione delle sessioni, in particolare al **Non Session Manager**, quindi esplorare le funzioni che vengono messe a disposizione.

Creare connessioni

Adesso dedichiamoci alla creazione di uno scenario realistico, così da dimostrare come creare delle connessioni con JACK. Utilizzeremo Ardour per registrare i seguenti strumenti:

STRUMENTI E INPUT	
Voci maschili	Input 1
Voci femminili	Input 2
Chitarra acustica	Input 3
Basso acustico	Input 4
Chitarra	Usando l'amplificatore per chitarra virtuale Guitarix
Percussioni	La Drum machine sincronizzata su Hydrogen
Piano	Tastiera MIDI collegata all'ingresso MIDI con pianoforte virtuale

Server modulare

JACK permette di collegare e sincronizzare i vari programmi audio insieme, in modo da trarre vantaggio dai punti di forza di ciascuna applicazione. La regolazione delle sessioni può essere utilizzata per gestire e richiamare sincronizzazioni molto ramificate. Una suite di applicazioni modulari, per esempio, è **Non Suite** che comprende **Non Timeline**, **Non Mixer** e **Non Sequencer**. Questo è solo un esempio di pacchetto completo, ma niente vi impedisce di crearne uno incorporando in un set modulare le applicazioni che preferite.

Nella parte superiore sinistra dell'interfaccia di Ardour vedrete una finestra contenente un mixer. Ricordate poi che abbiamo anche Guitarix e Hydrogen in esecuzione. Notate quindi la riga evidenziata. Tutti i collegamenti sono già stati istituiti, comprese le connessioni con Guitarix e Hydrogen. Per realizzare queste connessioni, in Ardour fate click sui tasti di ingresso presenti a sinistra nella riga evidenziata. Questo permetterà di far comparire una finestra in cui è possibile selezionare gli ingressi. In questo caso abbiamo voluto indirizzare il secondo input dall'interfaccia audio sulla seconda traccia (voci femminili), in modo da selezionare **capture 2** tramite la relativa casella. Ripetete la stessa procedura per l'altra traccia fino a quando tutti i collegamenti sono impostati come preferite. Un aspetto particolare a cui fare attenzione è che tutti i collegamenti che stabilite in Ardour compariranno anche negli altri programmi sincronizzati. Questo avviene non perché siano le applicazioni a realizzare i collegamenti ma perché è JACK a occuparsene. A proposito di connessioni, vale la pena citare il JACK connection manager **Patchage**.
Come detto, si tratta di un software che utilizza i colori per identificare le porte: il blu per l'audio e il rosso per il MIDI. Tenete comunque presente che Patchage può essere un'ottima soluzione solo se avete poche connessioni da gestire. In caso contrario, potreste essere sopraffatti dal disordine. Quando i collegamenti sono tanti, quindi, è meglio usare il gestore incluso in Ardour. Vi consigliamo comunque di prendervi un po' di tempo per provare vari JACK connection manager, usando poi quello che meglio si adatta al vostro flusso di lavoro. Tenete presente che tutti i programmi di questo genere fanno esattamente la stessa cosa, anche se in modi leggermente diversi tra loro. In definitiva, abbiamo visto come compiere i primi passi con JACK, dimostrandovi quanto questo server sia flessibile e modulare. Potete cominciare con poco, senza tante pretese, fino a sfruttarlo per progetti davvero professionali. Tutto dipende dalla vostra fantasia e dal tempo che dedicherete a imparare l'uso avanzato di questo potente strumento. Ci vorrà un po', ma i risultati non mancheranno di certo. LXP

Costruire un ambiente di sviluppo virtuale con PHP

Andiamo alla scoperta di macchine virtuali e script PHP per creare il perfetto ambiente di sviluppo

PHP potrà non essere grazioso ma di sicuro è utile e tutto ciò che facilita lo sviluppo e l'apprendimento del PHP può solo essere positivo. Nel corso di questo articolo vedremo come realizzare una macchina virtuale completa, in grado di gestire pagine Web, eseguire script PHP e spedire mail. Vi mostreremo inoltre come clonare una macchina virtuale e trasferire il tutto su una macchina diversa indipendentemente dall'architettura del PC ospite, a 32 o 64 bit.
Per proseguire avrete bisogno di scaricare l'ultima versione di Ubuntu, da installare poi sulla nuova macchina Virtualbox. Anche se questo articolo si concentrerà su **Virtualbox** esistono delle alternative per la virtualizzazione come **KVM**, **VMware** e **Xen**. Le macchine virtuali spesso sono una soluzione conveniente e versatile. Per esempio, potete ospitare pacchetti come Rsync o altri metodi per sincronizzare file o database tra varie macchine virtuali o direttamente con altri server.

Installare Virtualbox

Il metodo per installare Virtualbox su macchine basate su Debian consiste nell'aggiungere una riga nel file **/etc/apt/sources.list** per puntare al repository del pacchetto. Potreste dover abilitare la virtualizzazione dal BIOS del vostro computer per far funzionare Virtualbox correttamente a 64 bit. Vediamo un'esempio di installazione. Aprite il file **sources.list** con un editor di testo. Tutti i comandi devono essere eseguiti come utente root:

```
vi /etc/apt/sources.list
```

Aggiungete questa riga:

```
deb http://download.virtualbox.org/virtualbox/debian quantal contrib
```

E lanciate questi comandi:

```
wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo apt-key add -
apt-get update
apt-get install virtualbox-4.3
```

e confermate l'installazione. Se doveste avere problemi potete dare un'occhiata alla wiki ufficiale, all'indirizzo **http://bit.ly/VirtualBoxDLs**.



› **La creazione di una nuova macchina virtuale**

Con installazioni più datate, **sources.list** può essere un problema. Alternativamente, potete sempre fare un backup e aggiungere temporaneamente il sorgente di Virtualbox, o potete aggiornare il vostro **soruces.list**. Potete sempre generarne uno nuovo all'indirizzo: **http://repogen.simplylinux.cf**. Per aprire Virtualbox vi basterà cercarlo nel Dash o lanciarlo direttamente da riga di comando con **virtualbox** e per creare una nuova macchina, selezionate **Nuova**, scegliete il tipo, la versione e cliccate su **Avanti**. Fatto ciò, dovete decidere quanta memoria dedicarle o accettare quanto proposto. Potete cambiare questo valore anche in un secondo momento. Proseguite e scegliete di creare un disco fisso virtuale. Vi verrà chiesto il tipo di disco fisso: VDI è una buona scelta, oltre che quella di default. Proseguite rispondendo alle ovvie domande e completate la creazione. Di default per Ubuntu viene proposto l'utilizzo di un disco da 8 GB, potrebbe essere il caso di aumentare un pochino lo spazio disponibile. Con la macchina virtuale pronta ci basterà cliccare su **Impostazioni ➟ Archiviazione ➟ Vuoto** (sotto a Controller: IDE) ➟ **Scegli un file** (cliccando sul piccolo CD a destra) ➟ **Aprire l'iso** e cliccare **OK**. Ora possiamo avviare la nostra macchina, cliccando su **Avvia**. Potete procedere con l'installazione come in un normalissimo PC, nel caso di Ubuntu bisognerà selezionare **Installa Ubuntu**. Anche se sembrerà che l'installazione stia lavorando su un hard disk reale, sta semplicemente scrivendo all'interno del file .vdi da noi creato. Una volta conclusa l'installazione potrete avviare la macchina virtuale e fare login con il nome utente e password che avete creato. Il menu **Visualizza** offre diverse opzioni. In una nuova installazione di Virtualbox la visualizzazione potrebbe non essere ottimale. La buona notizia è che il problema può essere risolto con semplicità. Installando i **Guest Additions**. Potete installarli semplicemente andando nel menu **Dispositivi,** selezionando **Inserisci l'immagine del CD delle Guest Additions...** e seguendo le istruzioni. Una volta installati i Guest additions e riavviata la macchina virtuale modificando la finestra lo schermo si adatterà di conseguenza e sarà possibile scegliere tra le visualizzazioni **Schermo intero, Modalità trasparente** e **Modalità scalata**.

Cloni e configurazione server

Per clonare una macchina virtuale ci basta scegliere la macchina dall'elenco, cliccarci sopra con il tasto destro e scegliere **Clona**. Qui ci basterà dare un nome al clone e seguire i passi. Ma come fare per spostare il clone su un nuovo PC? Un modo per farlo è quello di spostare l'intera directory della macchina virtuale presente nella directory Virtualbox VMS. A quel punto basterà lanciare Virtualbox nella nuova macchina e scegliere **Macchina ➟ Aggiungi**. Un altro metodo è quello di creare una nuova macchina e collegarla al file VDI esistente. Ora che Virtualbox è configurato, vediamo di installare nel server anche **Apache**, **MySQL**, **PHP** e **Postfix** Ci vorrà solo qualche

minuto. Per installare Apache lanciamo il comando:

```
apt-get install apache2
```

come utente root. Dopo aver installato Apache, aprite un browser all'indirizzo localhost. Dovreste vedere la pagina di default di Apache2 per Ubuntu. Per MySQL, lanciamo:

```
apt-get install mysql-server mysql-client
```

sempre come utente root. Durante l'installazione verrà chiesta la password per l'utente root di MySQL. La password può essere lasciata in bianco, in quel caso non verrà creata ne richiesta in seguito, è possibile comunque cambiare questa impostazione di MySQL in un secondo momento, con un semplice comando di una riga. Ora non ci resta che installare PHP con il comando

```
apt-get install php5 libapache2-mod-php5
```

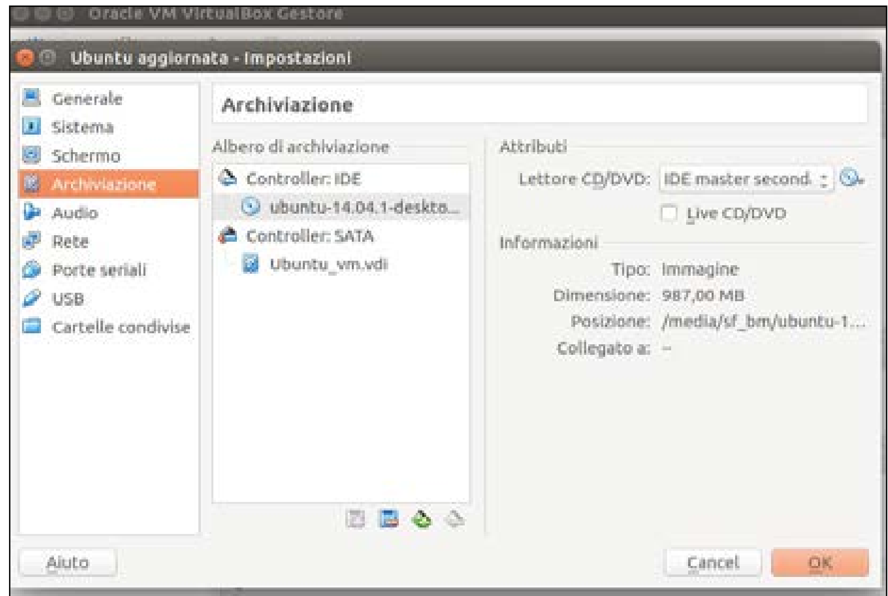
sempre come root. Al momento della stesura di questo articolo, la cartella di default utilizzata per il sito Web è **/var/www/html**. In versioni precedenti la directory di default era **/var/www**. Da notare inoltre che Apache imposta l'opzione **AllowOverride a None**: dovrete cambiarlo a **All** se avete intenzione di utilizzare i file **.htaccess**. Proseguiamo creando una pagina PHP di test per verificare che funzioni correttamente. Si può utilizzare qualunque editor per farlo. Di default, sono installati **Nano** e **Gedit**. Se preferite **vim**, dovete eseguire sempre come root questo comando:

```
apt-get install vim
```

Ora create un file chiamato **test.php**. Ovviamente, il file va posizionato all'interno della directory **/var/www/html** menzionata prima. All'interno del file inserite una semplicissima riga di codice:

```
<?php echo phpinfo(); ?>.
```

Questa funzione mostra tantissime informazioni relative al vostro PHP. Aprite la pagina con il vostro browser e vedrete i dettagli della vostra configurazione. Il file relativo alla configurazione si trova nella directory **/etc/apache2/php.ini**. Potete accedere come root e modificare questo file a piacere. Mentre PHP necessita dell'installazione, altri linguaggi di scripting come Python e Perl sono già installati di default. Un semplice **whereis python** e **whereis perl** vi darà conferma di ciò.



› **Scegliamo l'iso che vogliamo installare nella nostra macchina virtuale**

DNS e port forwarding

A questo punto, il server funziona e necessita soltanto di un paio di modifiche per essere visibile a un indirizzo tipo **esempio.it**. Se volete utilizzare un dominio, dovrete impostare il DNS in modo che punti al vostro IP. Potete utilizzare un servizio DNS gratuito o il vostro servizio di hosting per farlo. Tecnicamente, bisogna creare un record A che punta all'IP che vi è stato assegnato dal vostro provider. Una volta configurato il DNS, dobbiamo configurare il nostro router, in modo da abilitare il port forwarding della porta 80. Di default, Apache e altri server utilizzano questa porta. Potreste però voler esportare anche altre porte, per esempio per inviare mail (porta 25), SSH (porta 22) e volendo FTP (porta 21). Tutte queste regole utilizzeranno lo stesso IP locale della vostra macchina virtuale. Ok, ma qual è l'IP locale della macchina? Quando costruite una macchina virtuale con Virtualbox, la vostra rete utilizzerà NAT e prenderà l'IP di default della connessione host. Comunque, potete impostare Virtualbox in modo che la macchina virtuale prenda il proprio indirizzo IP. Potete modificare questa impostazione in qualsiasi momento prima di avviare la macchina virtuale. Per farlo, **tasto destro ➟ Impostazioni ➟ Rete ➟ Connessa a:**. Connettersi con NAT e Rete Interna sono affidabili e veloci, oppure potete utilizzare Scheda con Bridge con eth0 cambiando la modalità promiscua in Permetti MV. Questo secondo metodo permette alla vostra macchina virtuale di utilizzare la vostra scheda di rete e di avere il proprio IP. Se lanciate **ifconfig** nei vostro desktop e nella macchina virtuale dovreste vedere due IP differenti. Il router dovrebbe raggiungere entrambi allo stesso modo. Potreste utilizzare anche una scheda wireless USB per collegare Virtualbox ma in questo tutorial ci concentreremo soltanto sulla Ethernet che non soffre delle problematiche che a volte il wireless presenta. Ora che avete un server funzionante e che mostra pubblicamente le vostre pagine Web, come un VPS affittato o un hosting dedicato di qualche provider, avete tutti i tool di base necessari per un sito Web. Adesso possiamo dedicarci allo studio del PHP.

PHP

Ora che tutto è pronto, potete aprire il vostro editor preferito e iniziare a scrivere un po' di codice PHP. Anche se PHP ha i suoi detrattori, »

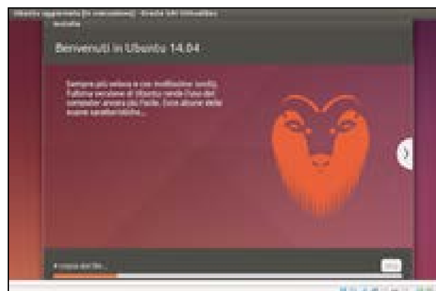
PHP e sicurezza

Nel vostro nuovo PHP installato, non c'è una modalità sicura ed esistono diversi problemi di sicurezza. Ma in un server domestico potete eliminare la maggior parte dei problemi disabilitando l'upload dei file e tutte le funzioni che accedono al sistema tramite Bash. Tutte le modifiche alla configurazione PHP possono essere eseguite nel file **/etc/php5/apache2/php.ini**. Per esempio, potete modificare l'impostazione **file_uploads** passandolo da **On** a **Off**. Di default, funzioni come **exec**, **shell_exec** e **base64** possono causare problemi se i file sono nel vostro sistema. Le funzioni come **shell_exec** ed **exec** vi permettono di lanciare comandi Linux che possono sovrascrivere file, mostrare dati utilizzando il comando **cat** o addirittura prendere il controllo dei database. Il Web è pieno di script pericolosi che possono mostrare e modificare i vostri dati. Questi non sono gli unici problemi di sicurezza del PHP. Codice scritto male può essere sfruttato per degli attacchi e file **PHP.ini** possono essere caricati nelle vostre directory. Questi file **PHP.ini** possono sovrascrivere il file principale. Questo problema può essere evitato utilizzando la patch **suhosin**. Nel complesso, il vostro server PHP casalingo sarà abbastanza sicuro se disabilitate l'upload dei file e non permettete l'accesso al vostro sistema. E ovviamente, non fidatevi mai dell'input inserito dagli utenti nelle form. Esistono diverse funzioni come **htmlentities()** e **mysql_real_escape_string()** per ripulire i dati inviati dagli utenti.

la sua popolarità è fenomenale e come linguaggio di scripting lato server è sicuramente il più diffuso. La maggior parte degli sviluppatori Web hanno il proprio sito ospitato su di una piattaforma con una pila LAMP (Linux, Apache, MySQL e PHP) o LEMP (**Nginx**, il server HTTP che sostituisce Apache, si pronuncia 'Engine-X', ecco perché E al posto di N). Questo ha fatto sì che nascessero moltissimi script e librerie che soddisfano moltissime necessità. Alcune sono applicazioni molto conosciute come Wordpress, Joomla, Drupal, Magento, Prestashop e molte altre. Oltre ai vari script ci sono molti framework che ci permettono di utilizzare delle librerie per scrivere codice, come Symphony, Codeigniter e Zend. Anche se PHP è comodo, è molto permissivo e quindi è molto facile scrivere del cattivo codice. Comunque, con un po' di attenzione potete scrivere oggetti solidi e codice robusto. Esistono diverse librerie che vi permettono di separare la logica dal design; come il template engine **Smarty**. Se conoscete C o Java, avete le basi per costruire codice robusto, e non avrete l'obbligo di dichiarare i tipi delle variabili. Esistono moltissimi script popolari in circolazione, purtroppo però molti hanno un lato oscuro. Alcuni sono molto grandi e sono diventati così popolari da essere diventati anche bersaglio dei cracker. Se si scrivono siti Web con codice PHP pulito, prestando attenzione alla sicurezza, il codice resta molto più gestibile e di solito è molto performante. Ci si impiega più tempo, ma con dedizione e pazienza potrete raggiungere il livello di conoscenza di PHP, MySQL, HTML e CSS che vi permetterà di costruire qualsiasi cosa possiate immaginare (con il dovuto tempo, ovviamente).

Lo scripting PHP

In questa sezione parleremo dello scripting PHP che può essere eseguito da riga di comando o dentro al browser. PHP viene interpretato dal server e mostra l'output al client. Sostanzialmente, è ciò che vedete ogni giorno dai siti Web che



› **Installiamo la nostra macchina virtuale esattamente come faremmo su di un normale hard disk**

utilizzano applicazioni PHP, come Wordpress e Magento. Potete creare script ed eseguirli da riga di comando, per esempio degli script di utility che potreste lanciare periodicamente utilizzando **cron** o manualmente al bisogno. Andiamo a vedere alcune funzionalità di base, come creare commenti, variabili, array, cicli, funzioni, classi e oggetti. Tutti i file PHP hanno come estensione .php. Tutto il codice all'interno dei tag **<?php ?>** viene interpretato. Tutto il codice all'esterno di questi tag viene considerato come HTML dal browser. Anche i tag **<? ?>** funzionano, ma solo se configurate il PHP (**php.ini**) per utilizzare gli short tags. In PHP esistono tre modi principali per inserire commenti (cioè delle note che non vengono interpretate in modo che possiate facilmente seguire cosa fa il codice).

```
// Commenti dopo le barre verticali
# Commenti dopo il cancelletto
/* I commenti tra questi caratteri possono essere
su più righe */
```

Le variabili semplici si dividono in stringhe (caratteri) e numeri e si dichiarano utilizzando il simbolo $. Nell'esempio qui sotto vedremo come dichiarare entrambi i tipi di variabile:

```
$mia_stringa = "Mia stringa";
$mio_numero = 3;
```

Potete stampare nella pagina racchiudendo il testo tra apici singoli o doppi. Gli apici singoli vengono trattati "come sono" quelli doppi invece vengono interpretati. Questo significa che potete stampare il contenuto delle variabili utilizzando i doppi apici.

```
echo 'Ciao!'; //output: Ciao!
echo 'Ciao $variabile'; //output: ciao $variabile
...con doppi apici:
$variabile = "Giovanni";
echo "Ciao!"; //output: Ciao!
echo "Ciao $variabile"; //output: Ciao Giovanni
```

Esistono casi in cui dovrete mettere dei doppi apici dentro altri doppi apici, come quando si usa l'HTML. In questi casi, dovete utilizzare la barra rovescia per far sì che non vengano interpretati. Vediamo un esempio

```
echo "<a href=\"esempio.it\">Mio collegamento
</a>"
```

Un'alternativa è quella sostituire i doppi apici con apici singoli, oppure quello di cambiare radicalmente approccio:

```
<?php ?>
<a href="esempio.it">Mio collegamento</a>
<?php //Aggiungere php qui
?>
```

Esempi di funzionalità

Chi proviene da C o Python, ha familiarità con la stampa e la funzione **sprintf()**. Un esempio del suo utilizzo è mostrato qui sotto:

```
$variabile1 = 10;
$variabile2 = 5;
$variabile3 = $variabile1 + $variabile2;
$variabile4 = 'squadra di calcio';
$formato = 'Ci sono %d uomini e %d donne per
un totale di %d persone nella %s.';
echo sprintf($formato, $variabile1, $variabile2,
$variabile3, $variabile4);
```

Gli array sono gruppi di oggetti. Ecco la sintassi degli array con indici e di quelli associativi:

```
$indicizzato = array('mela','pesca','pera','prugna');
$associativo = array('nome' => 'Giovanni', 'anni'
=> 32, 'altezza' => '1,85 metri');
```

Due comandi sempre utili sono **print_r()** e **var_dump()**. Vi mostrano le chiavi e i valori degli array se avete bisogno di dare un'occhiata più approfondita. Provate **print_r($indicizzato)** e **print_r($associativo)** per confrontare le coppie chiave-valore dei due array. PHP possiede molte funzioni per gli array che possono essere usate per ordinarli, aggiungere dei valori e molto altro. Le variabili e i blocchi di testo sono separati da dei punti che vengono chiamati concatenazioni. Vedrete l'utilizzo delle concatenazioni nella sezione dedicata ai cicli (di seguito). I tipi di ciclo più comuni sono il **foreach**, il **for** e il **while**. Normalmente, il ciclo foreach viene usato per scorrere un array. Potete separare un array in valori o in coppie chiave-valore. Vediamo di seguito alcuni esempi di diversi cicli e funzioni.

» **Ciclo Foreach** Questo tipo di ciclo scorre un array in ordine.

Scripting PHP e riga di comando

Quando eseguite uno script PHP da riga di comando viene interpretato in modo leggermente diverso rispetto a quando viene visto su di un browser. Una delle differenze principali è che per fare un invio nella riga di comando dobbiamo digitare **\n** mentre per un browser dobbiamo utilizzare il comando HTML **
**. Inoltre, quando si scrive codice PHP da lanciare tramite riga di comando bisogna aggiungere in testa al file:

```
#!/usr/bin/php
```

Poi può essere eseguito in questo modo:

```
php ./vostroscript.php
```

Vediamo una tipica pianificazione con **cron** che esegue il file **nomefile.php** ogni dieci minuti. Il file di configurazione di **cron** può essere gestito dall'utente root o da altri utenti all'indirizzo **/var/spool/cron/nomeutente** o semplicemente con il comando **crontab -e**.

```
*/10 * * * * nomeutente /usr/bin/php -f /var/
www/html/nomefile.php
```

I file che vengono eseguiti da **cron** possono richiedere alcune modifiche nel codice. I file eseguiti nel browser sono relativi alle cartelle **public_html** o **www**. Mentre nei file eseguiti con **cron** tutti i file inclusi o richiesti dallo script devono avere un indirizzo assoluto. Quindi un file incluso in questo modo "**miofileincluso.php**" funzionerà nel browser se si trova nella stessa cartella dello script, ma per farlo funzionare con **cron** sarà necessario utilizzare come percorso **/var/www/html/miofileincluso.php**. Un'alternativa alla modifica del percorso è quella di rimuovere tutte le inclusioni e inserire le funzioni necessarie direttamente all'interno del singolo file.

```
foreach ($indicizzato as $valore) {
        echo $valore."<br/>"; //concatenazione tra una variabile e il tag HTML <br>.
}
foreach ($associativo as $chiave => $valore) {
        echo $chiave." - ".$valore."<br/>"; // Concatenazione tra variabili e testo scritto.
}
```

» Ciclo for Questo ciclo For stampa il valore di $i che viene aumentato a ogni iterazione del ciclo. Continua finché il valore della variabile rimane inferiore a 5.

```
for ($i=1; $i<5; $i++) {
        echo $i; //output: 1234
}
```

» Ciclo while Questo ciclo while fa esattamente la stessa cosa del ciclo for appena visto

```
$i = 1;
while ($i<5) {
        echo $i; //output: 1234
        $i++;
}
```

» Includere file Se volete includere un file al file corrente, potete farlo utilizzando le funzioni **include()**, **include_once()**, **require()** e **require_once()**. Ecco un esempio:

```
include("miofile.inc");
```

» Funzioni Le funzioni sono insiemi di procedure che potete riutilizzare. Una funzione può essere invocata attraverso il suo nome. Normalmente, le funzioni si trovano all'interno di file e vengono incluse quando servono. Vediamo ora una semplice funzione personalizzata **scrivilo()**:

```
function scrivilo() {
        echo "Ecco il mio testo";
}
scrivilo();
```

PHP include migliaia di funzioni. Alcune di queste funzioni sono object-oriented, per esempio **new DOMDocument()**. **http://php.net/** è il sito dove andare per ogni necessità legata a PHP. All'indirizzo **http://php.net/manual/it/index.php** troviamo il manuale che è quasi interamente tradotto anche in italiano. Quando si utilizzano le funzioni PHP bisogna ricordarsi che utilizzano della memoria e in certi casi se ne usano moltissime. Se puntate alle performance utilizzare classi e oggetti può essere un'idea migliore.

» Classi e Oggetti Con PHP, potete scrivere programmaticamente dall'alto al basso. Ma, l'OOP (*Object Oriented Programming,* programmazione orientata agli oggetti) vi da un modo molto più efficiente per gestire e scrivere il codice.
La classe principale può avere delle classi figlie che sono in grado di ereditare alcune o tutte le proprietà della classe padre. Per utilizzare l'approccio OOP bisogna creare una classe che include metodi e proprietà. Una proprietà è simile a una variabile mentre un metodo è come una funzione. Per istanziare una classe si utilizza la keyword **new**. Proprietà e metodi possono essere pubblici, privati o protetti. Pubblici significa che sono accessibili da qualsiasi parte, privati che sono disponibili soltanto all'interno della classe e protetti che sono disponibili a figli e genitori delle classi nelle quali sono dichiarati. Un altro aspetto dell'OOP è l'utilizzo della keyword $this (questo termine viene utilizzato per fare riferimento alla classe stessa. Si utilizza poi **->** per accedere ai metodi). Il semplice esempio che segue dovrebbe chiarire questo concetto:

```
Class Prova {
        public $nome = 'Matteo';
        function prendi_nome() {
                return $this->nome;
        }
}
```

› Phpinfo mostra la configurazione del nostro PHP, possiamo cambiarla modificando il php.ini nella directory di default oppure creando un nuovo file php.ini all'interno della directory del nostro html

```
$mia_prova = new Prova();
echo $mia_prova->prendi_nome();
```

» Query MySQL Una funzionalità molto utile è la possibilità di scrivere query MySQL e scorrere i dati che ritornano. Le query possono essere eseguite utilizzando PDO o la funzione **mysql_query()**. Per eseguire una query è necessario essere connessi a un database.

» CRUD È l'acronimo di *Create*, *Read*, *Update* e Delete (Creazione, Lettura, Aggiornamento, Cancellazione). La maggior parte delle applicazioni Web utilizzano dei CRUD, scritti in un modo o un'altro, per permettere agli utenti di fare delle modifiche. Gli eventi di un CRUD utilizzano i comandi MySQL **SELECT**, **INSERT**, **UPDATE** e **DELETE**.

» Eseguire PHP dentro Bash Dato che ci sono buone possibilità che siate esperti nello scripting Bash potete sfruttare questa abilità aggiungendo del codice PHP all'interno di script bash con il tag Here Document ('EOF').

```
#!/bin/sh
php_cwd=`/usr/bin/php << 'EOF'
<?php echo getcwd(); ?>
EOF`
echo "$php_cwd"
```

Ora avete le basi per scrivere programmi PHP nella vostra nuova macchina virtuale. Potete farne un backup e caricarlo ovunque vogliate. Oltre ai backup di Virtualbox potete utilizzare **ssh**, **rsync**, **mysqldump**, **scp** e **cron** in modo da poter aver sempre dei backup pronti su altre macchine... non si sa mai. Molti siti in PHP e applicazioni Web utilizzano file e database. Avere un piano per clonare o mantenere sincronizzati questi dati dalla vostra macchina virtuale può essere molto utile e facile da implementare. Buon divertimento.. LXP

Server email

Per iniziare con le email, dovrete installare un server email. Ne esistono diversi disponibili come **Postfix** e **Exim**. In questo tutorial useremo Postfix. Per installare Postfix, lanciate il comando che segue.

```
root# apt-get install postfix postfix-mysql dovecot-core dovecot-imapd dovecot-lmtpd dovecot-mysql dovecot-pop3d
```

Durante l'installazione vi verrà chiesto di creare un certificato SSL, poi dovrete confermare la configurazione di postfix (premete **tab** per selezionare **ok** e **invio** per proseguire) scegliete **Sito internet** ➠ date un nome al mail server oppure lasciate quello di default. Una volta conclusa l'installazione, dovrete modificare il file **/etc/postfix/main.cf** e aggiungere uno script tipo Squirrelmail per gestire la posta in partenza e in arrivo.

Configurare un server email può richiedere del tempo, e va oltre gli obiettivi di questo tutorial. Comunque, potete testare l'invio di una mail con i semplici comandi mostrati qui sotto. Per testare l'invio, copiate ogni riga seguita da invio. Quando arrivate al punto alla fine del corpo della mail aggiungetelo seguito da invio come fatto per gli altri comandi. Non è un errore, è richiesto per inviare la mail.

```
telnet localhost 25
HELO mail.esempio.it
MAIL FROM: prova@esempio.it
RCPT TO: destinatario@dominio.it
DATA
Subject: Oggetto qui
Corpo del messaggio qui
.
QUIT
```



Premiata Amministreria Dottor Brown

Dr Chris Brown
Il Dottore si occupa di formazione, scrittura di articoli e consulenze su Linux. Trova che il suo *PhD* in fisica delle particelle non sia di alcun aiuto in questo tipo di lavoro.

Tecniche esoteriche per i sysadmin direttamente dai recessi più impenetrabili della sala server

Andy Tanenbaum

È giunto il momento di rendere omaggio ad Andy Tanenbaum. No, non è morto, ma sta per andare in pensione dopo 43 anni alla Vrije University di Amsterdam. Perché Tanenbaum si merita una citazione in una rivista dedicata a Linux? Beh, è stato Minix, il sistema operativo scritto da lui come strumento didattico, che ha ispirato Linus Torvalds a risparmiare per comperarsi un PC su cui farlo girare e che poi ha fornito il modello per Linux. Anche i suoi libri sono stati fonte di inspirazione.
In **Rivoluzionario per caso - Come ho creato Linux (solo per divertirmi)** Linus scrive "il libro chi mi proiettò verso nuove altezze fu **Sistemi operativi: progettazione e implementazione** di Andrew Tanenbaum". Ho una copia del suo **I moderni sistemi operativi**. Non lo apro da parecchio tempo, ma ricordo che quando l'ho terminato ero convinto di essere in grado di scrivere il mio sistema operativo, se solo avessi avuto tempo. Tanenbaum ha criticato con una certa franchezza la progettazione del kernel di Linux e ha avuto una discussione piuttosto, ehm, vivace con Linus a proposito dei vantaggi dei *microkernel* (vedere **http://bit.ly/LinuxIsObsolete** per la discussione originale e **http://bit.ly/ReliableOS** per un aggiornamento). Cosa insolita per un professore di informatica, Tanenbaum cura anche il sito Web **http://electoral-vote.com/**.
Dubito che Andy legga questa rubrica, ma gli auguro una lunga e felice pensione.

Identificare i browser

Il nostro desiderio di rimanere anonimi sul Web è messo a rischio dalla nostra tendenza a essere unici

I cookie sostanzialmente consentono a un utente di dire "Ciao, rieccomi qua" quando visita di nuovo un sito Web. I cookie di terze parti sono invece un po' più insidiosi, dato che permettono di seguire i movimenti di un utente tra i vari siti. Ovviamente è possibile configurare il browser in modo da rifiutare i cookie, ma la maggior parte dei siti moderni non funziona senza. E anche disabilitando i cookie è possibile usare altri sistemi per identificarvi, anche se non con il 100% di certezza. Un trucco sorprendentemente efficace consiste nel calcolare le impronte digitali (*fingerprint* in inglese) del browser raccogliendo informazioni offerte spontaneamente come l'intestazione *User Agent*, la lista dei plug-in, la lista dei font installati, il fuso orario e così via. In termini di teoria dell'informazione ognuna di queste fonti aggiunge qualche bit di entropia e tutte insieme fanno 30 o più bit, generando un'impronta digitale che, più spesso di quanto si possa pensare, è unica.
Un progetto lanciato dalla **Electronic Frontier Foundation** fornisce prove convincenti di questo. Se non mi credete andate su **http://panopticlick.eff.org** e fate click sul pulsante **TEST ME**. Usando quella che ritenevo una combinazione piuttosto comune di browser e sistema operativo (Internet Explorer 10 su Windows 7) ho scoperto che l'impronta digitale del mio browser risultava unica tra i 4.449.076 testati finora.
Il resoconto mostra anche quanta entropia genera ciascuno dei test delle caratteristiche del browser (vedere l'immagine qui sotto). Per i dettagli c'è **https://panopticlick.eff.org/browser-uniqueness.pdf**.
Nel documento vengono citate anche altre tecniche non implementate dal progetto che presumibilmente potrebbero incrementare la probabilità di un precisa identificazione: cose come le impronte digitali della pila TCP/IP, di Microsoft Active X e delle API di Silverlight, l'ordine in cui il browser spedisce le intestazioni HTTP e "un'ampia gamma di sottili test comportamentali di Javascript".

Caratteristiche del browser	bit di informazioni identificative	Uno in x browser ha questi valori
User Agent	10,65	1607,94
HTTP_ACCEPT Header	4,09	17.02
Detagli dei plug in	14,65	25717,79
Time Zone	2,65	6.28
Dimensioni schermo e colori	8,99	506,97
Font di sistema	2,57	5,93
Cookies abilitati?	0,43	1,35
Test Supercookie	0,91	1,88

› Ecco i dettagli dell'entropia dell'impronta digitale del mio browser, divise per categoria. Si tratta di Firefox 31.0 su Ubuntu 14.04

Il significato di un nome

Se vi state chiedendo da dove arrivi il nome *panopticlick* dovete sapere che un *panopticon* è una costruzione circolare con "celle" lungo il perimetro e un punto di osservazione centrale che consente a un singolo sorvegliante di osservare tutti gli internati. Il progetto intendeva creare un sistema di sorveglianza centralizzato in istituzioni come prigioni, ospedali, scuole e manicomi. Nella mitologia greca Panoptes era un gigante con cento occhi.

Le gioie di X

Impariamo come separare un'applicazione dalla sua interfaccia utente usando l'architettura client/server del sistema X Window

Negli ultimi numeri di **Linux Pro** mi sono occupato di interoperabilità tra Linux e Windows. In particolare il mese scorso ho esaminato VNC come strumento per accedere a un desktop remoto. Questo mese voglio parlare di una funzionalità spesso ignorata di X Window che permette di accedere a una applicazione grafica remota, compreso l'intero desktop, in una maniera del tutto differente: mi riferisco alla sua architettura client-server. Nel mondo Linux tutte le applicazioni grafiche usano X Window System. Le applicazioni vere e proprie (che si tratti di una calcolatrice con quattro funzioni, di un gestore di file o di un browser Web) sono i client. Ognuno di essi si collega ad un server X per avere l'accesso allo schermo, alla tastiera e al mouse di cui hanno bisogno per la loro interfaccia utente. Il protocollo che client e server usano per comunicare è chiamato semplicemente *the X protocol*, il protocollo X (effettivamente suona molto bene come titolo di un film apocalittico di fantascienza anni '70). Il server gira localmente (sulla macchina davanti alla quale siete seduti), il client può essere locale o remoto.

Dov'è il mio server?

Molti hanno difficoltà a comprendere questa configurazione perché sono abituati all'idea che in una relazione client/server il server sia la parte remota (il server Web, il server di posta e così via) mentre il client è la parte locale. In questo caso è il contrario. Il server è sempre locale: è il client che può essere remoto. Se però vi chiedete in cosa consiste in realtà il servizio fornito la definizione ha senso: il servizio è pensato per consentire l'accesso allo schermo dell'utente, alla tastiera e al mouse. Il sistema X Window nacque nel 1984 al MIT. Le versioni del protocollo si sono evolute velocemente da X1 (nel 1984) a X11 (1987) dove si sono fermate. Da allora abbiamo avuto X11R2, X11R3 fino ad arrivare a X11R7.7, la versione attuale. Fin dall'inizio uno dei principi guida è stato quello di fornire *mechanism not policy*, che significa che X non impone un Window Manager specifico o uno specifico *look and feel* per le applicazioni. Anche se in via di principio è possibile per un'applicazione assemblare i messaggi del protocollo X11 da inviare al server, nessuno scrive codice a questo livello: la libreria X (detta Xlib, risale al 1985) viene invece usata per incapsulare il protocollo X11. Stratificate al di sopra di essa sono emerse un certo numero di librerie di *toolkit*: GTK+, Qt, Motif, OpenLook e così via. Queste librerie definiscono l'apparenza dei vari controlli utente e quando si programma a questo livello inizia a manifestarsi un determinato stile di interfaccia. La scelta del window manager (Gnome, KDE, Xfce, Unity ecc.) influisce più sostanzialmente sull'aspetto e i loro relativi meriti sono oggetto di accesi dibattiti.
Per quanto riguarda il server X però si tratta semplicemente di client. Ricordate, X fornisce il meccanismo, non le linee guida.
I dettagli dei window manager e delle librerie di toolkit non sono importanti per la nostra storia questo mese. È sufficiente comprendere che un'applicazione grafica (un client X) si collega a un server X allo scopo di accedere alla sua interfaccia utente. Per sapere a quale server X connettersi un client X consulta una importante variabile di ambiente chiamata **DISPLAY**. Una variabile **DISPLAY** completa (raramente usata nella realtà) potrebbe presentarsi così:

```
DISPLAY=saturn:0.1
```

che dice al client X di collegarsi alla macchina *saturn* (è possibile usare anche un indirizzo IP) e usare lo schermo numero 1 del display numero 0.

Il server X nel corso del tempo

La prima implementazione di X sui PC, chiamata X386, risale al 1991. All'inizio era gratuita, ma si tramutò ben presto in un prodotto commerciale, cosa che causò l'avvio del progetto XFree86 nel 1992. Il suo nome era una specie di gioco di parole su X386. XFree86 rimase il server X preferito per Linux fino al 2004, quando alcune modifiche alla licenza del progetto fecero decidere alla Free Software Foundation che non era più compatibile con la GPL. La maggior parte delle distribuzioni Linux si ribellarono a questa situazione e passarono a X.org (un *fork* del progetto XFree86 avvenuto prima della modifica della licenza). X.org rimane tuttora il server X più diffuso nel mondo Linux.

Usare DISPLAY

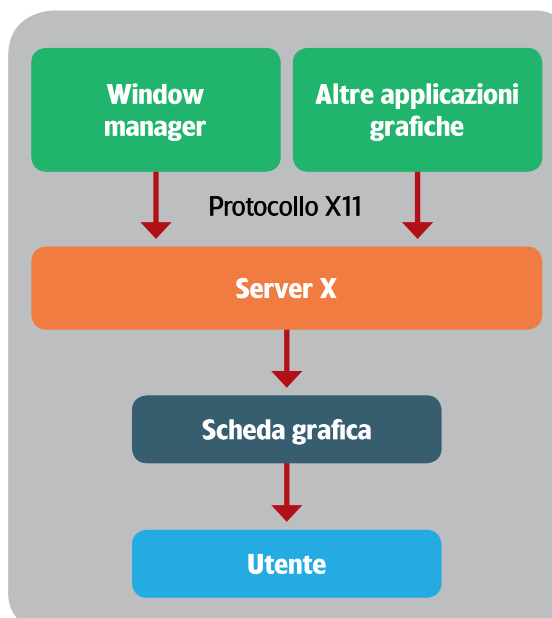
Un *display* sostanzialmente indica una tastiera, un mouse e uno o più schermi, in pratica qualcosa davanti a cui una persona può sedere e lavorare. Tutti i computer che ho visto (e anche le workstation di fascia alta) avevano un solo display in questa accezione del termine (non intendo un monitor fisico).
E la maggior parte dei display ha un solo schermo, il numero zero. Se non viene indicato assume il valore di default di zero:

```
DISPLAY=saturn:0
```

e se non viene indicato nemmeno il nome dell'host si arriva a

```
DISPLAY=:0
```

che dice al client X di connettersi al server sulla macchina locale, l'impostazione più comune della variabile.
Quando la connessione avviene con la macchina locale viene usato un socket di dominio UNIX (di solito **/tmp/.X11-unix/X0**). Quando »



› Applicazioni X Window (client) collegati a un server X locale per visualizzare la loro interfaccia utente

la connessione avviene con una macchina remota viene usato un socket TCP, di solito sulla porta 6000. I display aggiuntivi usano le porte 6001, 6002 e così via. La mia missione per questo mese (che potrebbe anche autodistruggersi in cinque secondi) è quella di far girare un'applicazione client X sulla mia Ubuntu box e mostrare la sua interfaccia sulla mia CentOS box. La macchina Ubuntu ha indirizzo IP 192.168.1.65 e la macchina CentOS 192.168.1.201. Dovreste riuscire a fare lo stesso con le vostre macchine (virtuali); in caso contrario prendetelo come un esperimento mentale, del genere di quelli che Schrödinger faceva con il suo gatto. Bisogna modificare un po' la configurazione lato CentOS. Innanzitutto è probabile che il server X sia configurato in modo da non restare in attesa di connessioni TCP. Per verificarlo provate questo:

```
ps -ef | grep X
```

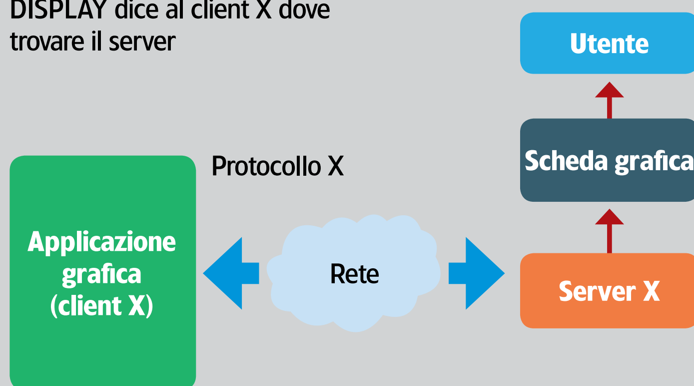
probabilmente comparirà l'opzione **-nolisten**. Analogamente il comando

```
# lsof -i TCP:6000
```

non produrrà probabilmente alcun risultato. Mr. Google mi dice che, per abilitare le connessioni TCP verso il vostro server, dovete aprire **/etc/gdm/custom.conf** e, nella sezione [**Security**], aggiungere la riga

```
DisallowTCP=false
```

Poi bisogna riavviare il sistema. Se ora ripetete i comandi **ps** e **lsof** provati prima, **ps** non mostra più l'opzione **-nolisten**, mentre l'output di **lsof** mostra che il Xorg è in ascolto sulla porta X11 (6000). Se tutto è a posto la prossima cosa da fare è aprire la porta 6000 sul firewall della CentOS box. Se usate lo strumento grafico di configurazione *system-config-firewall* fornito da CentOS/RedHat scoprirete che la porta 6000 non fa parte della lista standard dei servizi, quindi dovrete selezionare la voce **Altre porte** per aggiungerla. Infine dovrete disabilitare completamente il controllo degli accessi per il server X, cosa che si può fare con



› **Usando X le applicazioni possono girare su macchine diverse da quella su cui viene visualizzata l'interfaccia utente. Lo svantaggio è che la comunicazione attraversa la rete senza essere cifrata**

Perché X?

I nomi composti da una singola lettera sono molto diffusi negli ambienti informatici, anche se certamente lo spazio dei nomi possibili è limitato. Secondo Wikipedia ci sono linguaggi di programmazione chiamati B, C, D, E, F, G, J, K, L, Q, R, S e T. Così come il linguaggio C deriva dal precedente linguaggio B, il nome X deriva da un precedente sistema a finestre chiamato W, nato alla Stanford University verso il 1983. Girava su un sistema operativo a microkernel chiamato, ehm, V.

```
# xhost +
```

Questo comando abiliterà tutti i client a connettersi. È consigliabile invece abilitare semplicemente l'accesso dal sistema Ubuntu:

```
# xhost +192.168.1.65
```

A questo punto dovreste essere in grado di recarvi sulla vostra Ubuntu box, impostare la variabile di ambiente **DISPLAY** in modo che punti al vostro server X su CentOS e lanciare un client X (io ho scelto *gnome-calculator*):

```
$ export DISPLAY=192.168.1.201:0.0
$ gnome-calculator
```

Eseguire la Calcolatrice da remoto

Ed ecco... la calcolatrice sta girando sulla Ubuntu box, ma la sua interfaccia utente è su CentOS. Ovviamente avere un'applicazione la cui interfaccia utente sta girando su una macchina diversa da quella davanti a cui siete seduti non è l'ideale, quindi cerchiamo migliorare un po' le cose. D'ora in poi supporrò che avete un demone di *secure shell* (sshd) in esecuzione sul sistema Ubuntu e che su CentOS sia installato un client SSH. Tutto quello che dovrete fare è tornare sulla CentOS box, eseguire un login SSH su Ubuntu e poi lanciare il client X con l'interfaccia utente su CentOS. Il dialogo si presenterà più o meno così:

```
$ ssh chris@192.168.1.65
```

e dovrete inserire la password utente su Ubuntu. Sarete ora sul sistema Ubuntu:

```
$ export DISPLAY=192.168.1.201:0.0
$ gnome-calculator
```

A questo punto dovreste avere la calcolatrice in esecuzione su Ubuntu con la sua interfaccia utente su CentOS. Ovviamente *gnome-calculator* è un semplice esempio per illustrare quello che si può ottenere. Immaginate di sostituire la calcolatrice con un programma di simulazione meteorologica o con il software per il rendering della versione 3D di **Avatar 2** o con qualcos'altro ancora che brucia una quantità enorme di cicli di CPU. Immaginate poi di sostituire la semplice Ubuntu box con il supercomputer che si trova giù nel seminterrato dietro quelle porte blindate. Probabilmente questo esempio vi permette di comprendere meglio quanto sia utile e potente la capacità di X Window di separare l'applicazione dalla sua interfaccia utente. L'esecuzione di client X remoti presenta però alcuni svantaggi. Per prima cosa, come abbiamo visto, è un po' noiosa da impostare. In secondo luogo richiede di aprire una porta addizionale (la 6000) nel firewall. Terzo, e più importante, problema: è potenzialmente poco sicuro, perché il traffico relativo al protocollo X11 attraversa la rete non cifrato, permettendo a eventuali intercettatori di esaminare i dati trasmessi dall'applicazione e persino di iniettare per scopi fraudolenti eventi di mouse e tastiera.

Viene in soccorso la secure shell

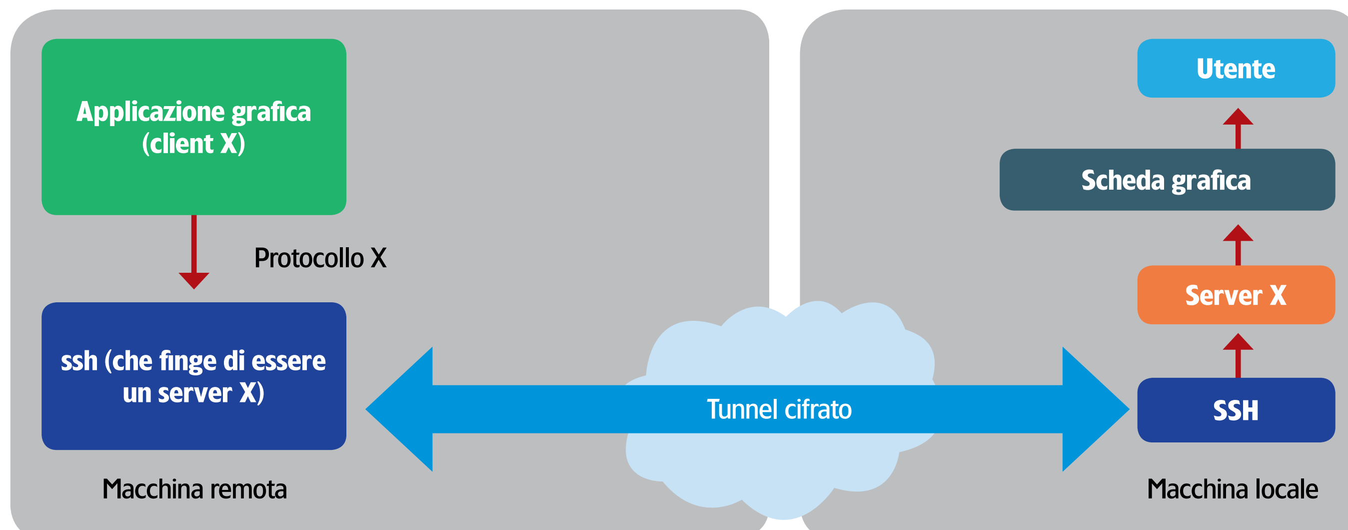
È a questo punto che viene in soccorso SSH. Come probabilmente saprete, SSH permette di stabilire un canale sicuro cifrato tra due macchine. Il suo scopo primario è quello di consentire un login a riga di comando, ma può anche essere usata per creare un *tunnel* al cui interno fare passare il traffico X11 e farlo è estremamente semplice. Per prima cosa bisogna assicurarsi che l'*X11 forwarding*, l'inoltro del traffico X11, sia abilitato sulla macchina su cui volete eseguire l'applicazione (nel mio caso la Ubuntu box). Verificate che la seguente riga

```
X11Forwarding  yes
```

sia presente in **/etc/ssh/sshd_config**. Ora, sul sistema CentOS, date semplicemente il comando

```
$ ssh -X 192.168.1.65 gnome-calculator
```

La magica opzione **-X** abilita l'inoltro del traffico X11: a questo punto



dovreste vedere la calcolatrice sul desktop CentOS. Come funziona tutto questo? Un semplice esame rivela quello che accade dietro le quinte. Su Ubuntu date il comando

```
$ sudo lsof -i | grep 6010
sshd    1459 chris 9u IPv4 154241 0t0 TCP localhost:6010
(LISTEN)
sshd    1459 chris 10u IPv4 154280 0t0 TCP localhost:6010-
>localhost:48291 (ESTABLISHED)
gnome-cal 1460 chris 6u IPv4 154278 0t0 TCP localhost:48291-
>localhost:6010 (ESTABLISHED)
```

Notate come la porta 6010 sia in ascolto (LISTEN) in attesa di connessioni e come sia anche connessa con la porta 48291, che appartiene al processo 1460, che a sua volta esegue *gnome-calculator*. Come mai *gnome-calculator* si è connesso alla porta TCP 6010 del suo server X? Esaminiamo l'ambiente del processo:

```
$ cat /proc/1460/environ
```

L'output del comando è piuttosto difficile da decifrare, più che altro perché mancano i ritorni a capo ma se guardate con attenzione noterete la definizione

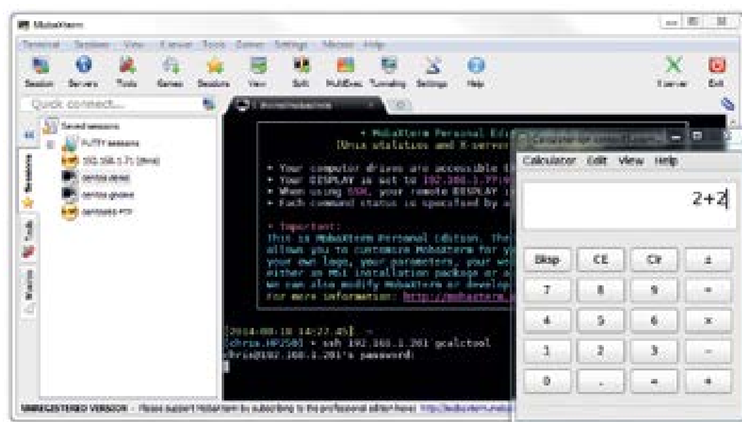
```
DISPLAY=localhost:10.0
```

Ecco la magia. Il demone SSH è in ascolto sulla porta 6010, fingendo di essere un server X. Ha anche impostato la variabile **DISPLAY** del processo di **gnome-calculator** in modo che si connetta al display numero 10 su localhost, cioè alla porta 6010. Il demone **sshd** spedisce poi il traffico al client SSH sul sistema CentOS, che a sua volta lo inoltra al suo server X locale. Non e richiesta alcuna modifica alla configurazione del server X su CentOS per far funzionare la cosa, può anche essere stato lanciato con l'opzione **-nolisten** abilitata. E non c'è bisogno di aprire porte in più sul firewall. Inoltre c'è l'ulteriore vantaggio della cifratura del traffico. Si tratta di una di quelle rare occasioni in cui si riesce a ottenere una maggiore sicurezza *e* una maggiore comodità: di solito bisogna rinunciare almeno in parte all'una o all'altra.

Non deve per forza essere Linux...

Finora ho dimostrato la natura client/server di X Window usando Linux sia per il client che per il server. Esistono tuttavia numerose implementazione di server X per Windows. Come ho scritto il mese scorso, **Xming** sembra essere la più diffusa; c'è poi un server X distribuito all'interno del toolkit **Cygwin**. Esistono anche implementazioni commerciali come **Exceed** e **X-Win-32** ma un'applicazione chiamata **Mobaxterm** ha recentemente attratto la mia attenzione. Mobaxterm è un po' come Cygwin: mette a disposizione degli utenti Windows una serie di strumenti simil-Linux a riga di comando e una vista stile Linux del filesystem di Windows. L'attenzione di Mobaxterm però è posta soprattutto sugli strumenti per l'accesso remoto. Supporta, tra gli altri, SSH, Telnet, rsh, VNC e FTP. C'è anche un server X11 e per default è abilitato l'inoltro X11. Nell'immagine qui sotto vedete un'applicazione X remota (**gcalctool**) in esecuzione sul mio sistema CentOS la cui interfaccia utente compare su Windows. Nell'ipotesi che i login via SSH funzionino non è richiesto alcun tipo di configurazione aggiuntiva. L'esecuzione remota di applicazioni X con inoltro SSH risulta pertanto davvero semplice. LXP

› SSH permette alle applicazioni X remote di accedere a una interfaccia utente locale usando un tunnel cifrato sicuro



› Una simpatica piccola applicazione, Mobaxterm, consente di usare un'ampia gamma di protocolli per l'accesso remoto e facilita enormemente l'inoltro su SSH di X Window

Desktop remoto

Portando all'estremo l'idea di avere un'interfaccia utente remota è possibile applicarla all'intero desktop. Sul mio sistema CentOS, per esempio, il comando **gnome-session** fa partire una nuova sessione di destop. Quindi dal mio sistema Window, usando di nuovo Mobaxterm, il comando

```
ssh 192.168.1.201 gnome-session
```

farà partire un intero desktop Gnome.

Magia! Anni fa tenevo un corso presso l'ufficio del catasto di Plymouth. Usavano una workstation Sun per fornire desktop grafici Unix separati a circa una dozzina di utenti, ognuno dei quali faceva girare un server X sul PC che si trovava sulla sua scrivania. La cosa interessante è che il limite di 12 utenti derivava dalla disponibilità di memoria sulla macchina Sun e non dalla disponibilità di cicli di CPU.

Costruisci il tuo NAS

Ecco come realizzare un NAS da zero con un po' di sano fai da te



Mano a mano che avete sempre più bisogno di spazio e di condividere i contenuti su tutti i dispositivi collegati alla LAN, la necessità di un NAS diventa impellente. A oggi esistono numerose scelte di modelli e prezzi che possono soddisfare le esigenze di chiunque. La maggior parte dei NAS sono disponibili con due o quattro dischi fissi, così da fornire una soluzione flessibile e completa per ogni necessità. Il cuore di molti NAS è proprio Linux che si dimostra uno dei migliori sistemi con cui configurare questo genere di dispositivi. A questo proposito, esistono alcune distribuzioni appositamente pensate per funzionare con i NAS. Tra queste, l'esempio più popolare è fornito da **FreeNAS** basato su BSD ma anche da **NAS4Free** e **OpenMediaVault** (un progetto basato su Debian, realizzato dall'autore di FreeNAS). Queste distro permettono di sfruttare un NAS in tutta libertà e mettono a disposizione strumenti già pronti per gli usi più comuni. Se però siete degli inguaribili smanettoni e non riuscite a fare a meno di pensare a un sistema fai da te, ci sono altre soluzioni a portata di mano. Basta pensare alla necessità di utilizzare il NAS come un moderno Media Center, oppure sfruttarlo per lo streaming dei giochi Steam da una finestra di Windows. Sono solo alcuni esempi di ciò che si può creare configurando un NAS in modo totalmente autonomo. Per realizzare lo scopo, in termini di hardware, dovete considerare il disco fisso come la parte più importante del progetto. In realtà basterebbe realizzare un semplice involucro con un unico disco di grandi dimensioni al suo interno, ma è sempre meglio investire qualche euro in più sulla ridondanza e quindi sul RAID. Non vi limitate a un solo disco fisso, ma acquistatene almeno due identici sia per prestazioni sia per dimensioni. In questo modo potrete sfruttare la modalità RAID1 (dove il secondo disco fa da immagine speculare dell'altro) o RAID5 (così da incrementare le prestazioni di scrittura). Nella nostra configurazione abbiamo voluto esagerare, usando ben quattro unità da 6 TB.

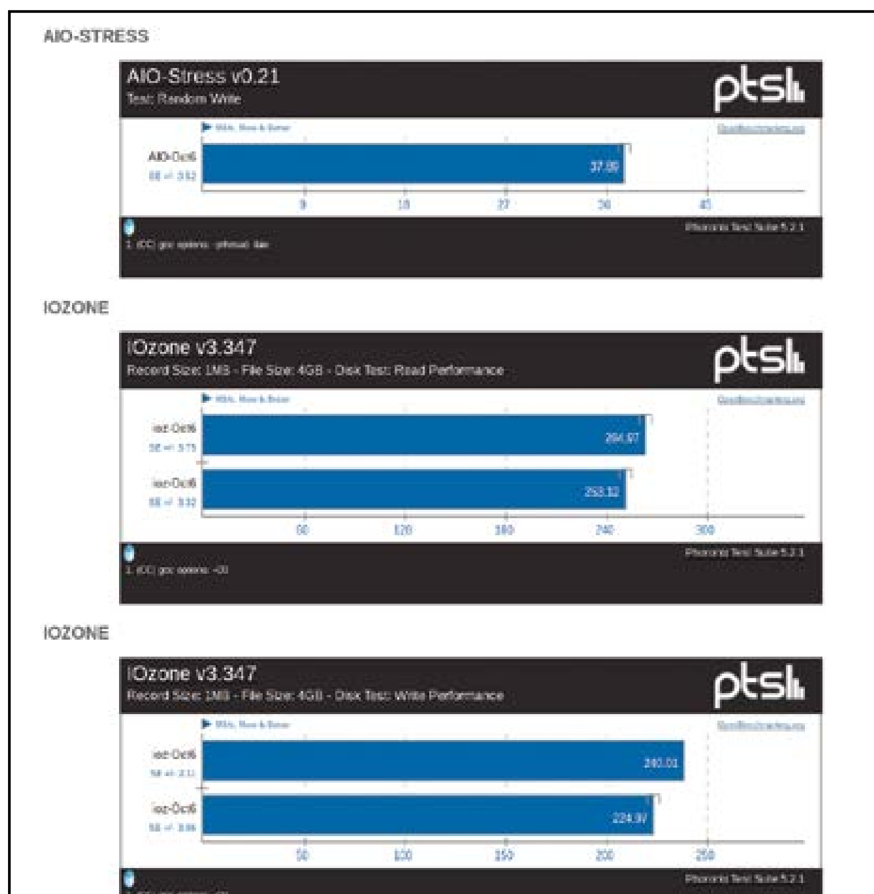
> **"Il cuore di molti NAS è proprio Linux che si dimostra uno dei migliori sistemi da configurare"**

Impostare il RAID

Con quattro dischi fissi esistono diverse soluzioni RAID da adottare. RAID 10, per esempio, è una combinazione delle modalità 1 e 0. In questo caso, viene prima di tutto impostata una modalità RAID0 per due dischi (non offre alcuna ridondanza, ma raddoppia le prestazioni), quindi si applica il mirroring dei contenuti sull'altra coppia di unità. Così facendo si ottiene il meglio in fatto di velocità e sicurezza. Sempre con quattro dischi è possibile sfruttare il RAID5, anche se si tratta di una modalità non sempre consigliata. Infatti, se un drive dovesse guastarsi, a causa dell'intensiva attività di ricostruzione, le probabilità di perdere un altro drive raddoppiano. RAID6, infine, può essere applicata a un minimo di quattro dischi fissi e grazie ai suoi vantaggi abbiamo deciso di adoperarla nel nostro esperimento. Questa modalità, infatti, fornisce un'assicurazione contro un doppio errore del disco e in più permette di incrementare le prestazioni grazie allo striping dei dati. Per quanto riguarda l'installazione del sistema operativo, ci sono un paio di considerazioni da fare. Sebbene sia possibile montarne uno su una partizione separata all'interno delle unità RAID, ci sentiamo di sconsigliare questa scelta. Infatti, dovreste ridimensionare tutte le partizioni di conseguenza. In realtà, il sistema operativo potrebbe essere installato all'interno dell'array, a condizione che il bootloader abbia una propria partizione e l'immagine **initrd** abbia un **mdadm** (*Multiple Administration Disk*) di sostegno. Tuttavia, anche in questo caso, abbiamo evitato di prendere in considerazione questa possibilità proprio a causa delle rigide condizioni richieste. Avendo quindi esaurito le opzioni a disposizione e con esse gli slot interni e le connessioni SATA, il nostro OS è andato a finire in un drive ibrido WD Black2 USB3. Questa soluzione va benissimo, a patto però di non rimuovere il disco esterno mentre il NAS è in funzione. Visto poi che per un server non avrete bisogno di un ambiente desktop in piena regola, abbiamo puntato su Arch Linux. L'idea di guardare ad Arch non renderà felici i neofiti ma dall'altra parte permette di sfruttare al massimo la versatilità di questo sistema. Grazie alla sua flessibilità, potete scegliere quali componenti installare e come configurarli fin dalla prima installazione. Se non sapete da dove cominciare, collegatevi alla pagina **http://bit.ly/ArchOnAUSBKey**, dove troverete tutte le istruzioni per installare Arch su un driver USB. Venendo poi alle funzioni con cui equipaggeremo il nostro NAS, una buona idea è quella di dotarlo di un server SSH, così come di un IP statico. In questo modo, potete scollegare tastiera e monitor e continuare a costruire il vostro progetto in remoto. Il primo passaggio da compiere è sicuramente il partizionamento dei dischi. Se le unità sono più grandi di 2,2 TB, allora avrete bisogno di utilizzare una tabella delle partizioni GPT. In questo caso, il programma **gdisk** è il nostro miglior amico ed è disponibile in Arch con il pacchetto **gptfdisk**:

```
# gdisk /dev/sda
```

Create una nuova partizione premendo **n**, quindi **Invio** per impostarla come primaria. Premete di nuovo **Invio** per accettare che contenga il settore di avvio di default [2048]. Di solito è una buona idea lasciare almeno 100 MB di spazio libero alla fine di ogni unità; dal momento poi che ogni drive deve avere la stessa capacità, è sufficiente fare qualche calcolo per capire il valore da impostare. Le partizioni RAID dovrebbero essere identificate con FD00, anche se in realtà Linux non presta particolare attenzione a questa dicitura. Quindi non vi rimane che scrivere la nuova tabella delle partizioni premendo il pulsante **W**. Ripetete questa operazione per tutte le unità che desiderate includere nell'array. »



› I nostri array alle prese con i benchmark AIO e IOZone

Scegliere i componenti

Fatta eccezione per i dischi, non è necessario preoccuparsi troppo per l'hardware. La macchina non deve essere potente; non c'è alcun bisogno di una scheda grafica dedicata e 4 GB di RAM saranno più che sufficienti per tutti gli usi. Le HP MicroServers sono una scelta molto popolare, anche se non rappresentano il meglio in fatto di eleganza. C'è però da dire che costruire tutto da soli dà una certa soddisfazione. Se avete un case micro ATX con annessa scheda madre, potreste creare qualcosa di interessante anche se non esteticamente molto piacevole.
Se la macchina che state per montare sarà collocata nel salotto di casa, avrete bisogno di componenti hardware silenziosi. Assicuratevi che il flusso d'aria interno sia buono, in quanto il surriscaldamento dell'hardware non è mai da prendere sottogamba. Noi abbiamo optato per un AMD Kabini 5350 APU (quad core, 2,05 GHz). La serie Kabini è in forte crescita, soprattutto nel settore a basso costo ed è stata lanciata nel mese di aprile. Dispone poi di un minuscolo TDP da 25W che non dovrebbe dare alcun problema di surriscaldamento. Il controller on-chip supporta nativamente solo due unità SATA, ma le schede dual port PCI-Express sono a buon mercato. Basta procurarsene una che consenta la commutazione basata su FIS. Se invece preferite Chipzilla, quindi il J1900 Celeron, possiamo dire che si tratta di una CPU sufficientemente buona e conveniente. In commercio, peraltro, si trovano parecchie schede Mini-ATX compatibili.

Configurare l'array

La parte più emozionante del nostro lavoro ma anche quella più impegnativa è la creazione dell'array. Gran parte della complessità è ovviata dal layer astratto **mdadm**, tuttavia è di primaria importanza assicurarsi di inserire i parametri corretti (le partizioni specificate, infatti, verranno cancellate irrimediabilmente):

```
# mdadm --create --verbose --level=6 --metadata=1.2 --chunk=256 --raid-devices=4 /dev/md0 /dev/sda1 /dev/sdb1 /dev/sdc1 /dev/sdd1
```

Il comando verrà eseguito in background per un periodo piuttosto lungo (noi abbiamo impiegato ben 24 ore per arrivare al termine). Tuttavia è possibile monitorare i progressi dando un'occhiata al file di stato con il comando:

```
# cat /proc/mdstat
```

Potete iniziare a utilizzare l'array in modalità *degraded* fin da subito ma la pazienza è una virtù e quindi vi consigliamo di mettervi l'animo in pace e trovarvi qualcos'altro da fare mentre aspettate che il processo termini. Tra i vari passatempi, suggeriamo di pensare al valore 256 K usato per la dimensione di ogni blocco. È stata davvero una scelta oculata? Il valore **chunk** si riferisce alla dimensione di ogni sezione di dati che vengono distribuiti sui dischi. Il fattore predefinito è 512 K, ma il valore ottimale dipende essenzialmente dall'hardware disponibile e dai casi. Per i file di dimensioni maggiori consigliamo di usare blocchi più piccoli, in modo che i dati si sviluppino su più unità. Per i file più piccoli, invece, si dovrebbero utilizzare blocchi più grandi. Se siete determinati a trovare il miglior valore possibile, allora dovrete mettere sotto torchio la vostra configurazione hardware per almeno un paio d'ore. Tenete poi presente che se state utilizzando una porta Gigabit Ethernet per accedere al NAS, è molto probabile incappare nel collo di bottiglia causato da questa interfaccia. Il valore che impostate, però, è importante quando il filesystem viene inizializzato. Dovete infatti descrivere l'array a mdadm, in modo che questo sia poi facilmente accessibile dopo il riavvio. Fatelo eseguendo il comando:

```
# mdadm --detail --scan >> /etc/mdadm.conf
```

che aggiungerà al file di configurazione mdadm

```
onveyance self-test routine
ecommended polling time:        (   5) minutes.
CT capabilities:              (0x303d) SCT Status supported.
                                       SCT Error Recovery Control supported.
                                       SCT Feature Control supported.
                                       SCT Data Table supported.

MART Attributes Data Structure revision number: 16
endor Specific SMART Attributes with Thresholds:
D# ATTRIBUTE_NAME          FLAG     VALUE WORST THRESH TYPE      UPDATED  WHEN_FAILED RAW_VALUE
  1 Raw_Read_Error_Rate     0x002f   200   200   051    Pre-fail  Always       -       0
  3 Spin_Up_Time            0x0027   206   203   021    Pre-fail  Always       -       8691
  4 Start_Stop_Count        0x0032   100   100   000    Old_age   Always       -       15
  5 Reallocated_Sector_Ct   0x0033   200   200   140    Pre-fail  Always       -       0
  7 Seek_Error_Rate         0x002e   200   200   000    Old_age   Always       -       0
  9 Power_On_Hours          0x0032   100   100   000    Old_age   Always       -       48
 10 Spin_Retry_Count        0x0032   100   253   000    Old_age   Always       -       0
 11 Calibration_Retry_Count 0x0032   100   253   000    Old_age   Always       -       0
 12 Power_Cycle_Count       0x0032   100   100   000    Old_age   Always       -       15
 92 Power-Off_Retract_Count 0x0032   200   200   000    Old_age   Always       -       5
 93 Load_Cycle_Count        0x0032   200   200   000    Old_age   Always       -       19
 94 Temperature_Celsius     0x0022   123   116   000    Old_age   Always       -       29
 96 Reallocated_Event_Count 0x0032   200   200   000    Old_age   Always       -       0
 97 Current_Pending_Sector  0x0032   200   200   000    Old_age   Always       -       0
 98 Offline_Uncorrectable   0x0030   100   253   000    Old_age   Offline      -       0
 99 UDMA_CRC_Error_Count    0x0032   200   200   000    Old_age   Always       -       0
 00 Multi_Zone_Error_Rate   0x0008   100   253   000    Old_age   Offline      -       0

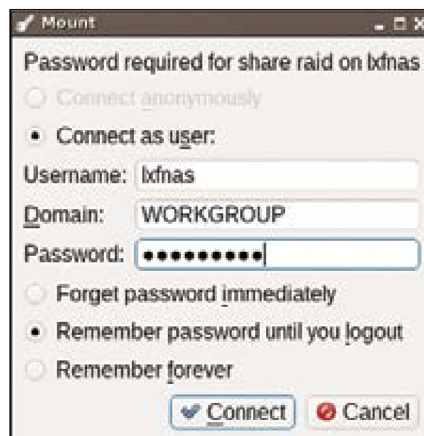
MART Error Log Version: 1
o Errors Logged
```

› Smartmontools può utilizzare i dati SMART e "profetizzare" eventuali problemi ai dischi

qualcosa di simile a:

```
ARRAY /dev/md0 metadata=1.2 name=wdarray:0 UUID=35f2b7a0:91b86477:bff71c2f:abc04162
```

Il device **/dev/md0** può essere trattato come qualsiasi altra partizione, anche se nel nostro caso si tratta di ben 12 TB. Iniziamo quindi a formattare la partizione per prepararla a ospitare i dati.



› La porta Qt di *PcManFM* è capace di gestire i supporti SMB. Si potrà poi impostare una password per l'accesso guest

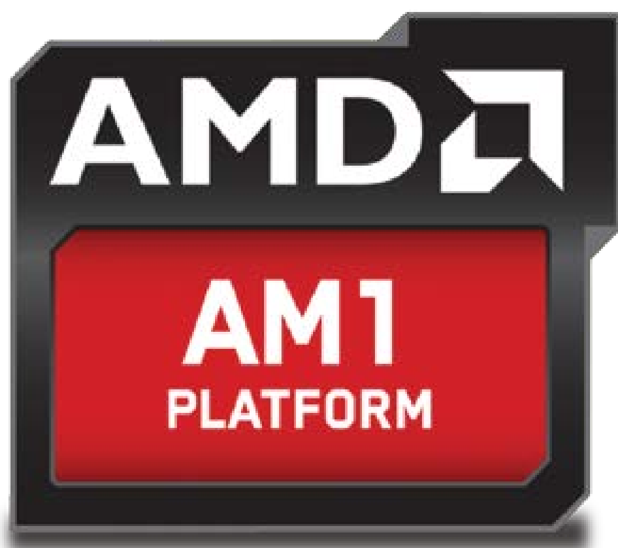
Noi utilizzeremo **ext4** che sebbene non sia un filesystem innovativo, consente di sfruttare al massimo solidità e affidabilità. Usare filesystem più moderni può sì permettere una migliore gestione delle unità ma è importante ricordare come SFS abbia bisogno di molta memoria e Btrfs possa causare picchi d'uso nella CPU. Ext4, da questo punto di vista, offre la migliore scelta possibile, in quanto trova nella configurazione RAID un perfetto connubio di funzionalità. Ext4 usa blocchi da 4k per impostazione predefinita (è comunque possibile usare blocchi più piccoli con l'opzione **-b**), in modo che ciascuno dei 256 k si estenda per 64 blocchi. Nella nostra configurazione a quattro dischi, ogni blocco viene suddiviso tra due dischi e gli altri due sono dedicati alla parità, in modo che la larghezza di banda sia di 128 blocchi. Se avessimo usato la modalità RAID5, solo un disco sarebbe stato un'unità di parità; tuttavia non è questo il nostro caso e così abbiamo formattato l'array con il comando:

```
# mkfs.ext4  -v -L wdarray -m 0.5 -E stride=64,stripe-width=128 /dev/md0
```

L'opzione **-m** imposta la percentuale della partizione da riservare al super utente. Questo valore è configurato in modo predefinito al 5%.

La verità sul RAID

Sicuramente avrete già letto quanto segue in qualsiasi altro articolo che parla di RAID, ma noi vogliamo darvi la nostra versione. RAID non deve essere confuso con backup! La modalità RAID è solo una prima linea difensiva, che però non protegge da cancellazioni accidentali e nemmeno da qualsiasi altra evenienza che porta alla perdita dei vostri dati. Se i file custoditi nei dischi fissi sono molto importanti, non c'è alcuna tecnologia che vi esenti dal creare backup giornalieri. Questo articolo vi mostra come configurare un sistema RAID, ma prima di tutto è necessario dissipare alcuni falsi miti sull'argomento. I controller RAID hardware dedicati sono sì disponibili ma nella maggior parte dei casi non sono necessari. È vero, nell'usarli c'è un piccolo guadagno a livello di elaborazione, ma sull'hardware moderno il vantaggio è quasi del tutto trascurabile. Un array RAID software può essere accessibile da qualsiasi sistema operativo Linux tramite il comando **mdadm**. I controller RAID hardware, invece, possono essere molto esigenti in fatto di compatibilità SATA. Infine, la scheda madre può pretendere il supporto per varie configurazioni RAID. Questo è conosciuto anche con il nome di FakeRAID o talvolta host RAID. Al di là del nome forse un po' dispregiativo (il controller onboard passa comunque tutti i calcoli RAID alla CPU), si tratta ancora di un setup robusto (anche se di solito supporta solo RAID 0 1 e 10). Questo vi permetterà di eseguire lo striping dell'unità di avvio e in alcuni casi anche di recuperare l'array dal BIOS. A volte, però, il recupero richiede l'utilizzo di software Windows. Ci dispiace molto ma purtroppo è vero.



› **La APU AMD Athlon 5350 è economica ed è abbastanza potente per mettere in piedi un buon NAS**



› **24 TB di qualità con la serie Red di Western Digital**

Balliamo la Samba?

Bene, a questo punto vogliamo aggiungere almeno un utente chiamato **lxpro**, quindi cambiare i permessi su **/mnt/raid** per consentirgli l'accesso ai nostri dati:

```
# groupadd raidusers
# useradd -m -G raidusers -s /bin/bash lxpro
# chown root:raidusers /mnt/raid
# chmod 775 /mnt/raid
```

Ora siamo in grado di avviare l'installazione di tutti i pacchetti richiesti con:

```
# pacman -S samba
```

Oltre ad avere un account di sistema, gli utenti Samba dovranno avere una propria voce nel file **smbpasswd**. Questo si può fare con il comando:

```
# smbpasswd -a lxpro
```

Ora è possibile modificare e leggere il file **/etc/samba/smb.conf**, in cui sono presenti diverse voci da abilitare o disabilitare. Per prima cosa decommentate e modificate la riga **hosts allow** in modo che l'accesso Samba sia limitato alla rete locale (per esempio **192.168.** o **192.168.0.**) o all'interfaccia di loopback. Verso la fine del file, troverete poi la sezione **Share Definitions**. Qui aggiungete il seguente blocco:

```
[raid]
    comment = LXP RAID
    path = /mnt/raid
    public = no
    valid users = lxpro
    writable = yes
```

È poi possibile aggiungerne altri, inserendoli nel gruppo raidusers, impostandone infine le autorizzazioni. Se si desidera configurare uno spazio pubblico che non richieda alcuna autenticazione, usate il comando:

```
# mkdir /mnt/raid/public
```

Per impostazione predefinita gli utenti *guest* non hanno privilegi di alcun genere e quindi, per dare accesso in scrittura, è necessario modificare il chmod di questa directory impostando il valore 777, infine rimuovere il commento dall'ultima riga in **Share Definitions**:

```
[public]
    comment = Guest area
    path = /mnt/raid/public
    public = yes
    read only = yes
    ;writable = yes
```

In alternativa, per consentire l'accesso ai soli utenti del gruppo **raidusers**, usate:

```
# chown lxpro:raidusers
```

> **"RAID non deve in nessun caso essere considerato sinonimo di backup"**

```
# chmod 775 /mnt/raid/public
```

quindi aggiungete la seguente definizione all'interno di [public] :

```
    writelist = @raidusers
```

Adesso siete in grado di avviare i servizi Samba e provare il server:

```
# systemctl start {smbd,nmbd}
```

Dovreste poter accedere alle condivisioni Samba da qualunque punto della rete.

Il servizio *mmdb* vi permetterà di controllare le condivisioni utilizzando le URI **\\hostname** (Windows) o **smb://hostname** (Mac/Linux). Se doveste avere problemi, provate con il vostro indirizzo IP. Sicuramente riuscirete nell'intento. Nautilus e Dophin vi permetteranno di sfogliare i gruppi di lavoro dalla sezione **Network**. Ricordate poi che le macchine in cui vorrete rendere visibili le condivisioni SMB avranno bisogno del pacchetto **smbclient** installato, così da navigare e montare le condivisioni di rete. Se non è possibile accedere alle condivisioni, allora sarà necessario prepararsi a risolvere qualche problema. Il comando **testparm** è sempre un buon punto di partenza, per poi proseguire nel controllo del file **smb.conf**, così da verificare la presenza di eventuali anomalie. Adesso sarà possibile impostare i vari servizi per l'avvio automatico sostituendo **start** con **enable**. Se volete montare automaticamente il server da un altro computer in rete, allora è possibile aggiungere il seguente comando al file **/etc/fstab**:

```
//192.168.133.225/raid  /mnt/raid      cifs
username=lxpro password=password 0 0
```

Memorizzare la password in questo modo non è ovviamente sicuro ma se prevedete di usare una macchina a cui accedete solo voi, può anche andare bene. Ricordate poi che se per qualsiasi motivo il servizio Samba non è in esecuzione, nel momento in cui verrà avviato su una macchina con **fstab**, noterete un rallentamento. »



› Le schede madri AM1 sono piuttosto limitate quando si tratta di fare i conti con le caratteristiche. Ciononostante vanno bene per i NAS, in quanto sono molto piccole

Aprirsi al mondo

Alle volte è necessario rendere il NAS accessibile da Internet. Tenete presente che questo passaggio non dovrebbe mai essere compiuto con leggerezza; soprattutto è qualcosa che non dovrebbe mai essere fatto con Samba. Un modo per eseguire un forward della porta 22 del router verso il NAS è connettersi via SFTP. Assicuratevi di aver bloccato il server SSH, in particolare la riga:

```
PermitRootLogin without-password
```

Questo limiterà l'accesso root alla sola chiave pubblica, quindi avrete bisogno di generarne una con **ssh-keygen** e aggiungerla al file **/root/.authorized_keys**. Per l'occasione potrebbe essere una buona idea disabilitare del tutto i log-in di root. Per farlo, impostate l'opzione di cui sopra con **no**. Questo, però, vi limiterà all'uso di un account root solo in locale e quindi dovrete per forza collegare un monitor e una tastiera al NAS. Dal momento che il vostro indirizzo IP esterno è destinato a cambiare ogni volta che ristabilite la connessione, è opportuno impostare un DNS dinamico utilizzando un servizio come **DuckDNS**, **dyndns** o **no-ip**. Questi servizi consentono di eseguire uno script o un programma client sul computer che aggiornerà il mapping DNS per l'indirizzo IP. Con la creazione di un cron job per eseguire questo software periodicamente, il NAS sarà sempre accessibile con un nome di dominio. L'iscrizione a questi servizi è molto semplice, tuttavia vi consigliamo DuckDNS che consente di aggiornare il tutto tramite un semplice script. Un'operazione, questa, in linea con la filosofia KISS di Arch. Per impostare DuckDNS seguite le istruzioni riportate nel box in fondo a questa pagina. Nella maggior parte dei casi, il vostro sistema RAID penserà a tutto ma in ogni caso è sempre meglio dare un'occhiata a **/proc/mdstat**. Una voce come [**UUUU**] significa che tutto sta funzionando a dovere. Se un disco dovesse guastarsi, al posto delle U troverete una fila di F. Tramite il comando che segue è poi possibile ottenere molte altre informazioni:

```
# mdadm --detail /dev/md0
```

Chi invece preferisce un approccio più manuale, potrà controllare regolarmente il proprio array. Qui potrete verificare le eventuali incongruenze tra i dati e i blocchi di parità. Per avviare una scansione, usate il comando:

```
# echo check > /sys/block/md0/md/sync_action
```

Non allarmatevi se il processo richiede parecchio tempo, è del tutto normale. Potrete comunque seguire le operazioni tramite il file **mdstat**. Se poi volete annullare l'operazione, potete farlo in qualsiasi momento lanciando l'istruzione:

```
# echo idle > /sys/block/md0/md/sync_action
```

Con un NAS, naturalmente, si può fare molto di più rispetto a quanto descritto in queste pagine, ma purtroppo abbiamo esaurito lo spazio a disposizione. Ci auguriamo quindi di avervi dato una buona infarinatura per cominciare a divertirvi in totale autonomia. LXP

> **"Talvolta può essere utile rendere il NAS accessibile direttamente da Internet"**

DuckDNS setup

È possibile iscriversi al servizio DuckDNS all'indirizzo **www.duckdns.org** usando Twitter, Facebook, Reddit o Google+. Dopo aver creato il vostro account, cambiate il nostro utente (lxpro) con il vostro e seguite le seguenti istruzioni:

```
# su lxpro
$ mkdir ~/duckdns
$ cd duckdns
$ nano duck.sh
```

Inserite quanto segue, sostituendo i domini e aggiungendo il token appropriato.

```
echo url="https://www.duckdns.org/update?domains=vostro_dominio&token=vostro_token&ip=" | curl -k -o ~/duckdns/duck.log -K -
```

Rendetelo eseguibile e provatelo:

```
$ chmod 700 duck.sh
$ ./duck.sh
```

A questo punto dovreste ottenere un output *curl*, quindi il file **duck.log** dovrebbe contenere l'incoraggiante **OK**. Per lanciare l'operazione automaticamente, è necessario installare un demone *cron*, per l'occasione usiamo il semplice **cronie:**

```
# pacman -S cronie
```

Il prossimo comando (ritornando al **lxpro user**) aprirà un crontab vuoto (a questo proposito vi consigliamo di usare l'editor **nano**):

```
$ EDITOR=nano crontab -e
```

Aggiungete la seguente riga per eseguire **duck.sh** ogni cinque minuti:

```
*/5 * * * * ~/duckdns/duck.sh >/dev/null 2>&1
```

ABBONATI SUBITO

SEI GIÀ ABBONATO?
RINNOVA ORA!
PER TE C'È UNO SCONTO
DEL 40%
4 NUMERI OMAGGIO

SCEGLI IL METODO PIÙ COMODO PER ABBONARTI:

•**ONLINE** sul sito www.linuxpro.it/abbonamenti

•**FAX** **invia il coupon** al N. 02 700537672

•**POSTA** Compila, ritaglia e spedisci il coupon in busta chiusa a: Sprea S.p.A. - Servizio Abbonamenti - Via Torino 51 - 20063 Cernusco Sul Naviglio (MI). Ti verrà inviato il bollettino precompilato a casa da pagare solo attraverso gli uffici postali.

•**TELEFONA** al N. 02 87168074 Dal lunedì al venerdì dalle ore 9,00 alle ore 18,00. Il costo massimo della telefonata da linea fissa è pari a una normale chiamata su rete nazionale in Italia. Via mail: abbonamenti@linuxpro.it

1 anno - 12 numeri

45,90€ invece di ~~70,80€~~

SCONTO 35%

ABBONANDOTI AVRAI DIRITTO AI SEGUENTI VANTAGGI

■ PREZZO BLOCCATO: per tutta la durata dell'abbonamento non pagherai un euro in più, anche se il prezzo di copertina dovesse subire aumenti.

■ TUTTI I NUMERI ASSICURATI: se per cause di forza maggiore qualche numero della rivista non venisse stampato, l'abbonamento verrà prolungato fino al raggiungimento dei numeri previsti.

■ RIMBORSO GARANTITO: potrai disdire il tuo abbonamento quando vorrai, con la sicurezza di avere il rimborso dei numeri che non hai ancora ricevuto.

Informativa ex Art.13 LGS 196/2003. I suoi dati saranno trattati da Sprea S.p.A., nonché dalle società con essa in rapporto di controllo e collegamento ai sensi dell'art. 2359 c.c. titolari del trattamento, per dare corso alla sua richiesta di abbonamento. A tale scopo, è indispensabile il conferimento dei dati anagrafici. Inoltre previo suo consenso i suoi dati potranno essere trattati dalle Titolari per le seguenti finalità: 1) Finalità di indagini di mercato e analisi di tipo statistico anche al fine di migliorare la qualità dei servizi erogati, marketing, attività promozionali, offerte commerciali anche nell'interesse di terzi. 2) Finalità connesse alla comunicazione dei suoi dati personali a soggetti operanti nei settori editoriale, largo consumo e distribuzione, vendita a distanza, arredamento, telecomunicazioni, farmaceutico, finanziario, assicurativo, automobilistico e ad enti pubblici ed Onlus, per propri utilizzi aventi le medesime finalità di cui al suddetto punto 1) e 2). Per tutte le finalità menzionate è necessario il suo esplicito consenso. Responsabile del trattamento è Sprea S.p.A. via Torino 51 20063 Cernusco SN (MI). I suoi dati saranno resi disponibili alle seguenti categorie di incaricati che li tratteranno per i suddetti fini: addetti al customer service, addetti alle attività di marketing, addetti al confezionamento. L'elenco aggiornato delle società del gruppo Sprea S.p.A., delle altre aziende a cui saranno comunicati i suoi dati e dei responsabili potrà in qualsiasi momento essere richiesto al numero +39 0287168074 "Customer Service". Lei può in ogni momento e gratuitamente esercitare i diritti previsti dall'articolo 7 del D.Lgs.196/03 – e cioè conoscere quali dei suoi dati vengono trattati, farli integrare, modificare o cancellare per violazione di legge, o opporsi al loro trattamento – scrivendo a Sprea S.p.A. via Torino 51 20063 Cernusco SN (MI).

Tagliare lungo la linea tratteggiata – Puoi anche fotocopiarlo per non rovinare la rivista

COUPON DI ABBONAMENTO

SÌ! Mi abbono a Linux Pro

Riceverò 12 numeri di Linux Pro a soli 45,90 € anziché ~~70,80 €~~ con lo sconto del 35%.

▹ **Inviate Linux Pro al mio indirizzo:**

Cognome e Nome

Via N.

Località CAP Prov.

Tel. email

▹ **Scelgo di pagare così:**

☐ Con il bollettino postale che mi invierete a casa

☐ Con carta di credito: ☐ Visa ☐ American Express ☐ Diners ☐ Mastercard

Numero

Scad. (mm/aa) Firma

▹ **Regalo Linux Pro a:**

Cognome e Nome

Via N.

Località CAP Prov.

Tel.

Compila, ritaglia e invia questo coupon in busta chiusa a:
Sprea S.p.A. - Servizio abbonamenti - Via Torino 51, 20063 Cernusco Sul Naviglio (MI)
Ti verrà inviato il bollettino precompilato a casa da pagare solo attraverso gli uffici postali

ABBONATI ANCHE SU INTERNET!
Collegati subito a: **www.linuxpro.it/abbonamenti**

Accetto di ricevere offerte promozionali e di contribuire con i miei dati a migliorare i servizi offerti (come specificato al punto 1 dell'informativa privacy): ☐ SI ☐ NO

Accetto che i miei dati vengano comunicati a soggetti terzi (come indicato al punto 2 dell'informativa privacy): ☐ SI ☐ NO

LXP150

OFFERTA VALIDA SOLO PER L'ITALIA

L'angolo di Android

News, recensioni e guide sul sistema operativo libero per smartphone

Se hai news da segnalarci o dei commenti scrivici ad angolo_android@linuxpro.it



Il 2015 sarà indossabile

L'annuale rassegna CES di Las Vegas ha mostrato poche novità nel settore degli smartphone, ma molte tra i dispositivi "indossabili"

Con il passare degli anni la tecnologia diventa sempre più "personale" e così nel corso dell'ultimo mezzo secolo si è passati dai grossi computer mainframe ai Personal Computer, ai portatili, allo smartphone per arrivare adesso ai dispositivi "indossabili". Il Computer Electronic Show (CES) di Las Vegas, che come ogni anno si svolge a gennaio, ha mostrato in anteprima le tendenze tecnologiche più interessanti proprio nel settore dei "wearable", cioè di quei dispositivi come braccialetti, orologi, ma anche capi di abbigliamento, che si portano addosso e ci forniscono informazioni di ogni tipo riguardanti noi stessi e chi ci sta intorno. In attesa che Apple presenti il suo smartwatch, a Las Vegas sono stati presentati modelli sempre più ricchi di sensori e allo stesso tempo esteticamente gradevoli, visto che ormai non sono solo oggetti del desiderio dei più geek. Tra gli smartwatch più interessanti c'è sicuramente quello realizzato da LG in collaborazione con Audi che entro il 2016 permetterà non solo di aprire e far partire l'auto con un tap ma anche di avere sempre sotto controllo il mezzo anche quando l'abbiamo parcheggiato. Garmin invece ha presentato con il **Fenix 3** l'evoluzione sportiva dello smartwatch. Un dispositivo cioè che grazie ai sensori integrati è in grado di adattarsi a ogni sport e fornire le informazioni su velocità, cadenza e condizioni fisiche. Tra le tante startup presenti a Las Vegas non ce n'erano molte di italiane, ma quelle poche si sono fatte notare, come è il caso dei ragazzi di Sensoria Fitness, **www.sensoriafitness.com**, che hanno avuto il coraggio di lasciare un lavoro fisso per dedicarsi alla realizzazione delle prime calze da corsa intelligenti, in grado cioè di fornire in tempo reale via bluetooth informazioni su come si corre e soprattutto su come si poggia il piede. E sono proprio i dispositivi rivolti al fitness quelli che stanno ottenendo il maggiore successo, come dimostra anche lo **Smart B Trainer** di Sony, uno speciale auricolare bluetooth che oltre ad avere 4 GB di memoria per ascoltare musica, integra anche un sensore GPS, un accelerometro e persino un rilevatore di battito cardiaco in modo che al termine dell'allenamento sia possibile scaricare sullo smartphone tutte le informazioni necessarie per valutare le proprie prestazioni.



› Al CES di Las Vegas Garmin ha presentato i nuovi smartwatch Fenix 3, in grado di adattarsi a ogni sport risultando allo stesso tempo esteticamente gradevoli



Telefono universale

Purtroppo capita spesso di trovarsi con lo batteria dello smartphone completamente a terra e con la necessità di dover fare una telefonata o mandare un SMS importante. Grazie a **Hotel My Phone**, una nuova app gratuita per Android, è possibile risolvere il problema utilizzando qualsiasi altro terminale Android e non necessariamente un telefono ma anche un tablet. In pratica dopo aver installato l'app sul proprio telefono verranno sincronizzati i contatti e i dati della SIM che poi sarà possibile recuperare da qualsiasi altro dispositivo Android su cui sia installata la stessa app, anche solo per quell'occasione. Sarà così possibile eseguire chiamate e inviare messaggi SMS sfruttando una connessione Voip e utilizzando il proprio numero telefonico. Il tutto senza far spendere un centesimo a chi ci presta lo smartphone, tranne naturalmente il costo di una eventuale connessione dati se manca il Wi-Fi. LXP

Motorola Nexus 6

Il primo Nexus realizzato da Motorola è il più grande e potente telefono Android mai costruito

A distanza di quasi tre mesi dalla presentazione, iniziano ad arrivare anche in Italia i primi **Nexus 6** realizzati da Motorola con l'ultima versione del sistema operativo Android 5.0 Lollipop (che nel frattempo è già diventato 5.0.2 e che secondo alcuni a breve diventerà 5.1). Rispetto ai modelli precedenti Nexus 6 si distingue sia per le maxi dimensioni da phablet (una via di mezzo tra smartphone e tablet) che per il prezzo da top di gamma (costa quasi il doppio del Nexus 5). Insomma ci sono tutti gli elementi necessari per accendere la curiosità degli appassionati, e siccome al momento Nexus 6 è disponibile quasi solo online dovrete giudicare leggendo le recensioni di chi lo ha già provato.

Grande dentro e fuori

La tendenza ormai affermata è quella di realizzare smartphone sempre più grossi ma Google con Nexus 6 ha voluto "strafare". Uno schermo da 6 pollici di diagonale è più simile a un tablet che a uno smartphone e non può essere tenuto con una mano sola anche da chi le mani le ha abbastanza grosse. Rispetto al Galaxy Note 4 e all'iPhone 6 Plus il Nexus 6 è solo leggermente più grande e pesante, eppure l'effetto è quello di avere tra le mani un terminale di un'altra categoria, più un tablet che un telefono considerando il peso di quasi 2 etti (per l'esattezza sono 184 grammi). Se poi a questo aggiungiamo un retro "morbido" e facilmente scalfibile che invita all'acquisto di una cover, si capisce come sia facile superare i 250 grammi complessivi. È anche vero però che la sensazione di avere tra le mani un piccolo mattone scompare non appena si accende il dispositivo e appare in tutto il suo splendore lo schermo Amoled con risoluzione QHD da 2.560 x 1.440 pixel, che può vantare una densità monstre di 500 pixel per pollice. Lo schermo è probabilmente il migliore oggi disponibile, Motorola ha fatto un ottimo lavoro anche grazie alla sua tecnologia Ambient Display dove i neri sono assoluti e non richiedono retroilluminazione, facendo così risparmiare energia. Se proprio dobbiamo trovare un limite, questo è costituito dall'impossibilità di variare la taratura cromatica che per alcuni potrebbe essere un po' troppo spinta, a meno naturalmente di non ottenere i privilegi di root e usare una delle tante app disponibili sul Web. Il vero problema resta comunque quello delle dimensioni perché Lollipop non ha alcuna soluzione in grado di adattarsi a un display da 6 pollici in modo da facilitare l'uso con una sola mano. La griglia della home ha icone troppo grandi e non permette di andare oltre alle 4 standard, a meno naturalmente di non usare un launcher alternativo, ma in questo modo si perde molto della esperienza Google Now.

> **"Una risoluzione Quad HD su uno schermo Amoled da 6" per godersi i video in 4K"**

Potenza e autonomia

Per fare girare un display ad altissima risoluzione senza blocchi o impuntamenti serve un motore potente e Motorola ha scelto quello che attualmente è il più veloce disponibile: il SoC Qualcomm Snapdragon 805 integra una CPU quad core da 2.700 MHz e un processore grafico Qualcomm Adreno 420 da 600 MHz che gestisce senza problemi anche i giochi 3D più complessi. Il tutto è accompagnato da 3 GB di memoria RAM e da 32 o 64 GB di memoria. Come da tradizione Nexus la memoria non è espandibile e così potrebbe essere una buona idea quella di spendere 50 euro in più per il modello da 64 GB. La batteria ha una capacità da 3.220 mAH che ci ha permesso di raggiungere la tarda serata anche con utilizzo molto intenso che ha compreso circa 4 ore tra navigazione, giochi e visione di video in streaming. La resa audio è ottima grazie ai due altoparlanti anteriori di buon livello, ottimo anche l'ascolto durante le telefonate sia in vivavoce che direttamente dalla capsula audio. La tecnologia LTE di ultima generazione (categoria 6) consentirebbe in teoria di navigare a 50 megabit al secondo ma questo naturalmente dipende dall'operatore e dal contratto. Personalmente non capiamo la necessità di usare una scheda Nano SIM in un dispositivo così grande, ma la tendenza in questo caso sembra essere quella che porta alla miniaturizzazione. La sezione fotografica è molto migliorata rispetto a Nexus 5, la fotocamera posteriore da 13 megapixel con stabilizzatore ottico di immagine e risoluzione dei video che arriva a 4K non ha veramente nulla da invidiare alla concorrenza e ripara una delle falle storiche dei terminali Nexus che hanno sempre avuto fotocamere scarse. In definitiva Nexus 6 è sicuramente uno dei più potenti e meglio costruiti dispositivi Android (e non solo) in commercio. Peccato che sia anche uno dei più costosi. LXP



› **Le dimensioni e il peso rendono molto difficile l'utilizzo del Nexus 6 con una sola mano**

Giudizio

Motorola Nexus 6

Produttore: Motorola
Web: **www.motorola.com**
Prezzo: 649 € (32 GB), 699 € (64 GB)

Caratteristiche	9
Autonomia	9
Prestazioni	9
Qualità/prezzo	7

» *Uno dei più potenti e migliori dispositivi Android in commercio. Peccato che sia tra i più costosi.*

Il voto di Linux Pro **8**

TRiO Pulsar 5

Uno smartphone pensato per chi non cerca le prestazioni ma vuole distinguersi dalla media senza spendere troppo

Nel corso del 2014 gli smartphone con display di grandi dimensioni, definiti anche phablet, hanno trainato il mercato dei dispositivi mobili. TRiO si è gettata nella mischia con il nuovo **Pulsar 5**, uno smartphone con display da 5,5" e una linea inconfondibile. Sveliamo subito il finale: il Pulsar 5 costa 239 euro; si colloca pertanto in quella fascia di "phablet economici" per la quale la domanda fondamentale è: a cosa dobbiamo rinunciare per spendere così poco (rispetto ai modelli top, ovviamente)? La prima sensazione, guardandolo per la prima volta, è quella di avere a che fare con un prodotto di fascia alta. La cover posteriore presenta un bell'effetto fibra di carbonio zigrinata antiscivolo, che ritroviamo anche sul frontale sotto il vetro dello schermo. L'intero bordo è impreziosito da una fascia di alluminio all'interno della quale, sul lato inferiore, sono integrate le griglie delle due casse stereo. Sul lato sinistro si trova il bilanciere del volume, ben raggiungibile anche con una sola mano, mentre su quello di destra troviamo il solo tasto di accensione. Sottile e leggero, 165 g in uno spessore di 7,6 mm, dà in mano una buona sensazione di solidità. Dal punto di vista del design, possiamo dire che sono soldi spesi bene, fermo restando il gusto personale. Passando all'analisi delle caratteristiche hardware il discorso cambia con risultati altalenanti. Il display è un IPS da 5,5" con risoluzione da 960 x 540 pixel, per una densità di 240 dpi. Il processore è un MediaTek MT6582 Quad core a 1.3 GHz supportato da 1 GB di RAM, mentre la memoria interna è di soli 8 GB, ma espandibile tramite slot microSD. La fotocamera posteriore è da 8 Mpixel mentre quella frontale da 1,9 Mpixel con flash LED. I numeri non dicono tutto quindi vediamo di analizzarli nella prova su strada. La risoluzione HD del display è decisamente bassa rispetto a quella dei modelli di fascia alta ma nell'utilizzo quotidiano si nota poco la differenza. Abbiamo fatto un giretto nelle gallerie di Flickr e le belle immagini in **Home** splendevano brillanti. Certo, messo a fianco del Sony Xperia Z3 la differenza è evidente, ma è davvero necessario? Dipende dall'uso che se ne fa. Discorso analogo vale per i film. Il display è grande e i video girano fluidi. Dal punto di vista delle prestazioni siamo nella media, la combinazione processore/RAM consente di svolgere senza problemi tutte le principali funzioni e fare anche un po' di multitasking senza impuntamenti o ritardi. Riesce anche a supportare degnamente un gioco di corse automobilistiche mediamente impegnativo come **Asphalt 8**. Le due casse stereo sono più belle che valide, vanno bene per ascoltare la musica di un video su YouTube o il parlato di un film, ma non pensate di usare il Pulsar come player MP3 predefinito. La memoria interna da 8 GB è poca. Tolti il sistema operativo e le app principali ne abbiamo a disposizione poco più di 4. L'inserimento di una scheda microSD è praticamente obbligatorio. Per finire, il comparto fotografico: anche in questo caso siamo nella media, forse un pochino sotto, con scatti solo accettabili, anche in buone condizioni di luce. L'aspetto che più ci ha deluso è però il sistema operativo. Il Pulsar 5 monta Android 4.2.2 Jelly Bean in versione standard. È una versione di Android decisamente datata che priva il Pulsar di alcune caratteristiche che avrebbero potuto fargli fare un'ulteriore scatto in avanti, come per esempio la nuova app **Fotocamera** di Kitkat. Un vero peccato. Tra le caratteristiche che invece gli fanno fare un passo in avanti rispetto a molti concorrenti, c'è il supporto per le due Sim. È una caratteristica che, in realtà, interessa un numero limitato di persone ma è comunque una funzione difficile da trovare anche nei modelli più costosi. LXP



› Se avete un numero aziendale e uno personale, e cercate la comodità di gestirli da un unico telefono, la doppia Sim (standard) del Pulsar 5 è decisamente quello che fa per voi

Scheda tecnica

- » **Sistema operativo** Android 4.2.2 Jelly Bean
- » **Processore** MediaTek MT6582 Quad core a 1.3 GHz
- » **Dimensioni** 154 x 79,2 x 7,6 mm
- » **Memoria** 1 GB di RAM
- » **Peso** 165 grammi
- » **Schermo** 5,5''
- » **Risoluzione schermo** 960 x 540 pixel
- » **Espansione** MicroSD

Giudizio

TRiO Pulsar 5

Produttore: TRiO
Web: **www.triohq.com**
Prezzo: 239 €

Caratteristiche	8
Autonomia	8
Prestazioni	8
Qualità/prezzo	8

» *Uno smartphone dal grande schermo che si distingue e costa poco.*

Il voto di Linux Pro 8

Recensioni

Tutte le novità in campo software e hardware testate e valutate ogni mese dai nostri laboratori

Se vuoi segnalarci qualche novità scrivi a recensioni@linuxpro.it

Una breve legenda

Ogni test di questa sezione è accompagnato da un giudizio che riassume con quattro indici numerici le principali qualità dell'applicazione o del prodotto hardware messo alla prova. I laboratori di Linux Pro assegnano un voto da 1 a 10 alle seguenti categorie:

Caratteristiche: fornisce tutte le funzioni di cui abbiamo bisogno? È innovativo?
Prestazioni: esegue in maniera efficiente le sue funzioni? È veloce e affidabile?
Facilità d'uso: dispone di un'interfaccia grafica chiara e facilmente fruibile? La documentazione che lo accompagna è sufficientemente completa ed esaustiva?
Qualità/prezzo: ha un prezzo competitivo? Vale i soldi richiesti per il suo acquisto?

Il nostro giudizio viene poi riassunto da un voto finale, espresso anche graficamente. Ecco la legenda dei voti:

10 Nulla da eccepire. Un prodotto praticamente perfetto.
8-9 Un buon prodotto. I pochi difetti presenti non sono gravi.
6-7 Compie il suo lavoro ma necessita di ulteriori sviluppi.
5-4 Deve migliorare prima di raggiungere un voto sufficiente.
1-3 Un completo disastro. Gli sviluppatori devono tornare alla fase di progettazione.

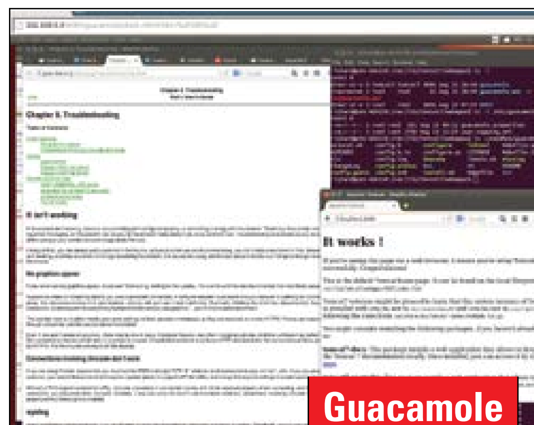
Ricordiamo infine che i software citati nelle sezioni Confronto e Da non perdere sono spesso presenti nel DVD sotto la voce "Rivista" sotto forma di codice sorgente o binario.

QUESTO MESE...


Tonido Raspberry Pi



i3



Guacamole

Tonido Raspberry Pi

Da utenti veterani di ownCloud, ci siamo domandati se in questo contesto vi sia spazio per un software proprietario: ecco la risposta...

In passato abbiamo recensito positivamente **Tonido** per Linux e ora questo server gratuito per la condivisione dei file è disponibile anche per il Raspberry Pi. Per chi non conoscesse questo software, Tonido vi permette di accedere ai vostri file e contenuti multimediali su un computer remoto. È inoltre disponibile in versione preinstallata su un piccolo plug computer con porte Ethernet e USB. La sua semplicità si estende anche alla versione per il Pi. Per farlo funzionare è sufficiente scaricare sul Pi l'archivio compresso di Tonido, estrarne il contenuto e avviare il server. Potrete quindi configurare Tonido tramite un browser remoto. Non dovete far altro che creare un ID unico per Tonido, che identificherà sul Web la vostra installazione di Raspberry Pi Tonido. Questo ID viene utilizzato per il servizio relay di Tonido, che vi permetterà di accedere al Pi e a tutti i file e cartelle condivisi da qualunque computer connesso a Internet, senza ulteriori configurazioni. Non dovrete più trafficare con firewall e servizi Dynamic DNS: vi basterà usare il Tonido ID specifico, dopo di che un breve wizard di configurazione vi chiederà di selezionare le cartelle a cui dovete accedere a distanza. Di default Tonido condivide tutte le cartelle ma potete personalizzare le impostazioni in modo da specificare manualmente specifiche cartelle. Vi verrà quindi chiesto di indicare le cartelle che contengono i vostri file musicali, fotografici e video. Potete anche modificare queste impostazioni dall'interno dell'interfaccia Web di Tonido. Una delle capacità più pubblicizzate di Tonido è quella di trasmettere contenuti multimediali in streaming. Dopo che avrete indicato le cartelle che contengono i file multimediali, Tonido effettuerà una scansione al loro interno e inserirà tutti i file multimediali rilevati nelle rispettive librerie virtuali, permettendone l'accesso rapido. Il software contiene inoltre un lettore integrato che è risultato in grado di leggere i file MP3 e MP4 ma si è bloccato di fronte ai file OGG e AVI. Gli sviluppatori consigliano di installare le librerie FFMPEG per ottenere un supporto multimediale più completo.



› **L'interfaccia di Tonido è molto intuitiva e comprende un lettore multimediale integrato**

Condividiamo?

Condividere una cartella è piuttosto semplice. Non dovete far altro che utilizzare il file manager integrato per selezionare una cartella locale sul Raspberry Pi (che può trovarsi anche su un disco USB collegato) e assegnarle un nome. Potrete inoltre applicare i permessi di condivisione del caso. Per esempio, potete fissare una data di scadenza per la condivisione, permettere a tutti coloro che dispongono dell'URL per la condivisione di accedere alla cartella o limitare l'accesso a pochi eletti. Quando limitate l'accesso a utenti specifici, potete anche controllare il diritto di ciascuno di loro di visualizzare e caricare file, nonché limitare la quantità di dati che ciascuno può caricare. Tonido conserva inoltre una cronologia relativa all'accesso ai file condivisi. L'unico limite della versione gratuita è che potete impostare un massimo di cinque condivisioni e di cinque utenti ospiti. Per caricare i file potete servirvi del File Manager integrato o semplicemente trascinare i file con il mouse nella finestra del browser. La possibilità di trascinare intere cartelle è disponibile solo per il browser Chrome. Non vi sono tuttavia barre di progresso per il monitoraggio dei caricamenti. Non c'è di fatto alcun indicatore che segnali che i file selezionati siano effettivamente in fase di caricamento. Inutile dire che questo è particolarmente scomodo quando si caricano file di grandi dimensioni. Per sincronizzare i file dovrete procurarvi un client. Tonido offre client per Windows, Linux e Mac; ma il client per Linux è disponibile solo sotto forma di file .deb ed è compatibile solo con i sistemi a 32 bit. Dopo averlo installato, potrete servirvi del client per sincronizzare i file con il Raspberry Pi, senza alcuna difficoltà. Potete utilizzare l'interfaccia Web per modificare la posizione della cartella sincronizzata. Nel complesso, Tonido è progettato per permettere alla maggior parte degli utenti di spiccare il volo verso il cloud. La sua utilità, tuttavia, comincerà ad assottigliarsi man mano che diventerete più esperti ed esigenti... LXP



In evidenza



Accesso universale

Il servizio relay TonidoID vi permette di accedere al vostro cloud da qualunque computer connesso a Internet.



Condivisione pubblica

Potete consentire l'accesso ai dati in remoto creando con facilità condivisioni private e pubbliche.

Giudizio

Tonido per Raspberry Pi

Produttore: CodeLathe LLC
Web: www.tonido.com
Licenza: Proprietaria

Caratteristiche	6
Prestazioni	6
Facilità d'uso	8
Documentazione	8

» *Un server per la condivisione dei file che facilita configurazione e amministrazione. Ideale per gli utenti inesperti.*

Il voto di Linux Pro **7**

SolidRun CuBox-i4Pro

La nuova versione del computer "cubico" è pronto a stupirvi!

Chi ci segue da un po' di tempo sa bene che apprezziamo particolarmente la piattaforma ARM e che spesso ci piace provare i vari miniPC (o schede simili) che sempre più spesso il mercato ci propone.
Per questo numero abbiamo avuto l'occasione di testare la nuova versione del CuBox (ne avevamo parlato parecchio tempo fa), in particolare il top di gamma **CuBox-i4Pro** di SolidRun.

OS a scelta

Questo miniPC dalla forma cubica è più piccolo di una pallina da tennis, giusto per darvi un termine di paragone. Nella parte frontale c'è il ricevitore infrarosso, sul fianco destro un'uscita audio ottica S/PDIF mentre nel lato posteriore trovano posto una Gigabit Ethernet (che però non viene sfruttata alla massima velocità ma è limitata a 470 Mbps), due USB 2.0, una eSata II, uno slot per microSD (per ospitare la scheda di memoria con il sistema operativo), una microUSB (da usare come console di sistema), il jack per l'alimentatore e una porta HDMI 1.4 (con supporto CEC). Internamente, invece, il miniPC è animato da un SoC Freescale i.MX6 Quad core (basato su Cortex-A9), con 2 GB di RAM e un processore grafico GC2000. All'elenco delle porte dobbiamo aggiungere che il CuBox-i4Pro dispone anche delle connessioni Wi-Fi e Bluetooth. Insomma, non manca proprio nulla. Prima di poter usare questo computer è necessario scrivere l'immagine del sistema operativo su una scheda microSD. Per il Cubox-i sono disponibili diverse distro Linux, oltre ad Android; potete vedere l'elenco all'URL **www.solid-run.com/wiki/Category:Ignition**. Per facilitare l'installazione di uno di questi sistemi, SolidRun mette a disposizione sul suo sito l'immagine di un installer unico (Ignition), che una volta avviato consente di scegliere il sistema da scrivere su SD. Nelle nostre prove con questo tool non siamo riusciti a installare Android 4.4.2 (lo abbiamo dovuto fare scaricando l'immagine) mentre hanno funzionato a dovere OpenELEC e Debian.
Con OpenSUSE abbiamo incontrato qualche difficoltà dopo il primo riavvio e abbiamo dovuto riscrivere il sistema usando la sua immagine.
L'installer si scrive su SD con un semplice comando dd:

```
sudo dd bs=4k conv=fsync if=ignition.img of=/dev/sdb
```

Fate attenzione, se provate, a indicare il corretto device di output (**of=/dev/sdX**).



› **Oltre al modello provato, il top di gamma, SolidRun offre altri tre modelli, più economici, interessanti per chi cerca solo un media center a basso consumo**

Impressioni d'uso

Il CuBox-i4Pro si dimostra molto versatile: lo si può usare come media center accanto al televisore, usando Android o XBMC, come server oppure come thin client. Lo si può collocare ovunque, viste le sue ridotte dimensioni, non scalda troppo neanche dopo molte ore di funzionamento, è silenzioso e ha prestazioni che vanno da dignitose, se usato come "desktop" (se non si avviano molti programmi assieme), a più che buone quando utilizzato come media center. Se si cerca un computer per navigare, mandare email, scrivere dei documenti e ascoltare un po' di musica (e non si intende giocare), il CuBox-i4Pro può anche sostituire il classico desktop senza troppi problemi. Il vero "difetto" al momento riguarda il software: alcune delle distro provate non hanno riconosciuto al volo tutto l'hardware (tipicamente Bluetooth o Wi-Fi), quasi tutte non sono riuscite a spegnere la macchina una volta dato l'opportuno comando (che in Android non c'è proprio), costringendoci a staccare la spina, non essendoci un pulsante fisico di accensione/spegnimento. Si tratta però, in gran parte, di difetti di gioventù: vista la buona community che frequenta il forum di SolidRun siamo sicuri che in breve tempo spariranno. Dopo qualche operazione di setup manuale, comunque, noi abbiamo usato per diverse settimane questo miniPC con buona soddisfazione. LXP

Benchmark

Benchmark	CuBox-i4Pro	Raspberry Pi B+	Penguin Wee
Blowfish*	49,00	102,00	5,28
Cryptohash**	21,75	8,61	302,07
Fibonacci*	11,80	26,78	1,61
N-Queens*	40,81	87,35	4,33
FFT*	49,43	154,18	0,96
Raytracing*	57,68	135,21	3,93

* il risultato minore è il migliore
** il risultato maggiore è il migliore

Giudizio

CuBox-i4Pro

Produttore: SolidRun
Web: **www.solid-run.com**
Prezzo: 139,99 $ (118,25 €)

Caratteristiche	**8.5**
Prestazioni	**8**
Documentazione	**8**
Qualità/prezzo	**8.5**

» *Quasi perfetto come media center, va bene anche come thin client/server a basso consumo per Internet.*

Il voto di Linux Pro 8.5

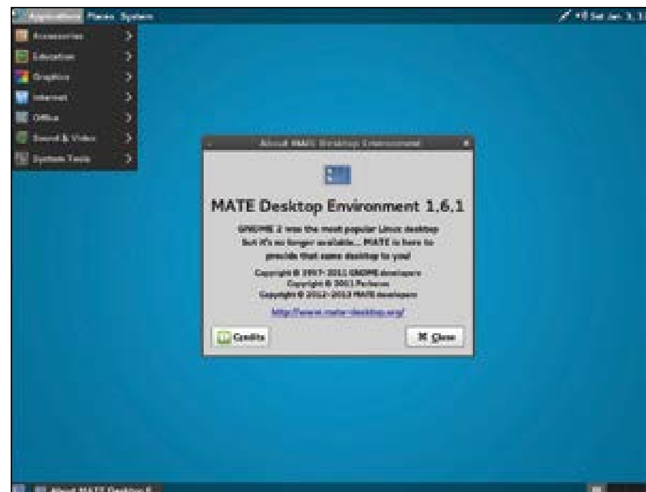
GhostBSD 4.0

BSD può essere annoverato tra gli acronimi che vengono utilizzati per richiamare la vera essenza dell'Open Source. Con GhostBSD abbiamo provato un sistema operativo alternativo

Il progetto **GhostBSD** è basato sul sistema operativo FreeBSD enterprise e comprende tutte le comodità che ci si aspetta da una distro desktop pensata per l'uso quotidiano. L'ultima versione, vale a dire la 4.0, è disponibile sia per macchine a 32-bit sia per quelle a 64-bit sotto forma di immagini ISO o IMG; rispettivamente per essere utilizzate con unità ottiche o periferiche USB. Le versioni precedenti di GhostBSD utilizzavano più desktop, ma la v4 sfrutta Mate con tre layout: un look classico con menu nella parte superiore e un pannello in basso; una versione minimalista; una disposizione semplice con un normale dock nella parte inferiore. All'interno del tradizionale menu che ospita i vari programmi, troverete applicazioni come Firefox, LibreOffice, Shotwell, Pidgin, Exaile e Gnome Mplayer. GhostBSD è straordinariamente reattivo. Si comporta bene anche su hardware piuttosto datato con 2 GB di RAM. Su macchine più recenti, invece, abbiamo dovuto fare i conti con il mancato supporto per UEFI di FreeBSD in Secure Boot, che purtroppo è ancora in via di sviluppo. Durante le prove, per avviare l'installazione di GhostBSD 4 abbiamo dovuto impostare il PC per l'avvio legacy. Questa release è la prima a includere il nuovo gestore pacchetti **pkgng** di FreeBSD. Si tratta di uno degli strumenti più veloci da utilizzare. Sfrutta repository che vengono definiti nel file **/usr/local/etc/pkg.conf**, il quale, a sua volta, contiene altre opzioni di configurazione. Purtroppo, allo stato attuale, pkgng non ha un'interfaccia grafica e quindi dovrete fare tutto da terminale.

Demoni al lavoro

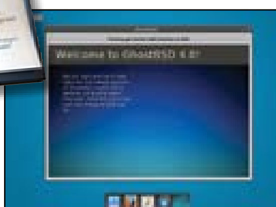
Dopo aver esplorato per qualche decina di minuti l'ambiente Live ci si sente subito pronti a installare il sistema sul disco fisso. Abbiamo a disposizione uno schema di partizionamento automatico che permette di prendere in consegna l'intera unità da gestire. Troviamo anche una funzione per il partizionamento manuale, da utilizzare se vogliamo far coesistere GhostBSD con un altro sistema operativo. Prima di selezionare questa opzione, tuttavia, è meglio passare un po' di tempo a familiarizzare con lo schema di partizionamento di BSD. Il programma di installazione offre anche la possibilità di installare il boot loader di GhostBSD. Un'operazione, questa, che si rende indispensabile se vogliamo fare di questo sistema il fulcro principale del nostro PC. Questo componente è poi in grado di gestire anche l'avvio di Windows. Se però prevedete di accoppiare GhostBSD con un'altra distro Linux sulla stessa macchina, consigliamo di evitare l'installazione del bootloader di Ghost, preferendo invece l'utilizzo del più tradizionale Grub 2. In questi casi, è sufficiente installare GhostBSD, quindi avviare l'altra distro Linux ed eseguire il comando **update-grub** per aggiungere Ghost al menu di avvio. Il progetto di questa distribuzione sta attraversando un momento di transizione. Attualmente supporta il desktop Mate ma ciò non significa che nelle versioni future possa esserci un'ulteriore svolta (lo sviluppatore sta cercando delle alternative come Xfce). Abbiamo trovato anche diverse modifiche al background che hanno avuto un impatto positivo sulla velocità e la reattività dell'intero sistema. Uno dei più grossi deficit che il progetto GhostBSD dovrà affrontare nell'immediato futuro è la mancanza di un front-end grafico per il sistema di gestione dei pacchetti. Questa situazione, infatti, potrebbe allontanare gli utenti meno affezionati e certo non attirarne di nuovi. A parte questo, tuttavia, non possiamo fare a meno di classificare GhostBSD come un ottimo sistema operativo desktop, capace di soddisfare chiunque voglia utilizzare il computer quotidianamente. È semplice, intuitivo e particolarmente veloce anche su macchine datate. In altre parole, una scelta alternativa ai nomi più blasonati del panorama delle distro Linux. LXP



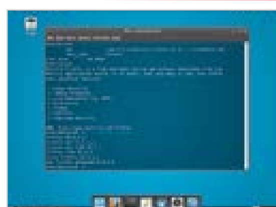
› **GhostBSD si comporta esattamente come una normale distro basata su Mate e permette quindi di sfruttare tutte le applicazioni più popolari per l'uso quotidiano**

In evidenza





Installazione facile
La grafica aiuta davvero molto durante il primo approccio.



Gestore pacchetti
Scordatevi la grafica, qui si fa tutto da riga di comando.

Giudizio

GhostBSD 4.0

Produttore: Eric Turgeon
Web: **www.ghostbsd.org**
Licenza: BSD

Caratteristiche	7
Prestazioni	8
Facilità d'uso	7
Documentazione	7

» *A parte il suo gestore pacchetti a riga di comando, si tratta di un sistema operativo molto versatile, che nessun utente Linux farà fatica a usare.*

Il voto di Linux Pro **7**



Se leggi Linux Pro, ti possono piacere anche...

OFFERTA SPECIALE
La guida completa ai droni: come sceglierli e come guadagnarci
COD. IMCS 2
€ 9.90



COD. RGV6
€ 9.90



COD. MICR1
€ 9.90



Completa la tua collezione ordinando gli arretrati a soli € 5,90 cad.
su www.linuxpro.it/arretrati
oppure utilizzando il modulo qui sotto

SCEGLI L'ARRETRATO CHE VUOI ORDINARE

SE VUOI ORDINARE VIA POSTA O VIA FAX, COMPILA QUESTO COUPON

Ritaglia o fotocopia il coupon, invialo in busta chiusa a: **Sprea S.p.A.** Via Torino, 51 20063 Cernusco s/n (MI), insieme a una copia della ricevuta di versamento o a un assegno. Oppure via fax al numero 02.700537672. Per ordinare in tempo reale i manuali collegati al nostro sito **www.linuxpro.it/arretrati**. Per ulteriori informazioni puoi scrivere a **arretrati@linuxpro.it** oppure telefonare allo 02/87158224 tutti i giorni dalle 14.00 alle 18.00

INSERISCI I CODICI delle pubblicazioni che desideri ricevere:

	€
	€
	€
	€
Totale Ordine	€

SCELGO IL SEGUENTE METODO DI SPEDIZIONE:
Indica con una ✗ la forma di spedizione desiderata

	Spedizione tramite posta tradizionale al costo aggiuntivo di	**€ 2,90**
	Spedizione tramite Corriere Espresso al costo aggiuntivo di	**€ 7,00**
	TOTALE COMPLESSIVO	€

Data

Firma del titolare

9 771722 616763 50150

NOME
COGNOME
VIA
N° **C.A.P.** **PROV.**
CITTÀ
TEL.
E-MAIL

SCELGO IL SEGUENTE METODO DI PAGAMENTO (Indica con una ✗ quello prescelto)

☐ Versamento su **CCP 99075871** intestato a **Sprea S.P.A. ABBONAMENTI Via Torino 51 20063 Cernusco Sul Naviglio MI** (Allegare ricevuta nella busta o al fax)

☐ Bonifico intestato a **Sprea S.P.A. Abbonamenti** sul conto **IBAN IT05 F076 0101 6000 0009 9075 871**

☐ **Carta di Credito** N.
(Per favore riportare il numero della Carta indicandone tutte le cifre)
Scad. CVV

Nome e Cognome del Titolare della carta (può essere diverso dall'abbonato)

☐ CartaSi ☐ postepay ☐ MasterCard ☐ VISA ☐ Diners Club ☐ American Express

Informativa e Consenso in materia di trattamento dei dati personali - (Codice Privacy d.lgs. 196/03) Sprea S.p.A. Socio unico Sprea Holding S.p.A. con sede legale in via Beltrami 21, 26100 Cremona, è il Titolare del trattamento dei dati personali che vengono raccolti, trattati e conservati ex d.lgs. 196/03. Gli stessi potranno essere comunicati e/o trattati da Società esterne Incaricate. Ai sensi degli artt. 7 e ss. si potrà richiedere la modifica, la correzione e/o la cancellazione dei dati, ovvero l'esercizio di tutti i diritti previsti per Legge. La sottoscrizione del presente modulo deve intendersi quale presa visione, nel colophon della rivista, dell'Informativa completa ex art. 13 d.lgs. 196/03, nonché consenso espresso al trattamento ex art. 23 d.lgs. 196/03 in favore dell'Azienda.

Gnome 3.14

Gnome si rinnova ancora una volta e per non lasciare delusi i suoi fan lo fa in modo migliore rispetto alle passate release. Le nuove caratteristiche abbondano e sono tutte ben riuscite

Prima dell'avvento delle tecnologie touch, sfondi, icone e barre delle applicazioni regnavano assolutamente sovrani. Nel corso del tempo, molti elementi sono stati modificati, aggiunti o rimossi ma le caratteristiche essenziali sono rimaste inalterate. Poi è arrivata la grande rivoluzione touch e Gnome ha deciso di passare dal tradizionale desktop a quello che oggi conosciamo come Gnome Shell. Questo passaggio a dir poco drastico ha portato molti ex-sostenitori ad abbandonare il progetto, andando a ingrandire la platea di chi sostiene alternative come Cinnamon di Mint. Nonostante le critiche, tuttavia, Gnome continua a innovarsi e lo fa con un sano mix di nuove funzionalità e molti miglioramenti, così da soddisfare vecchi e nuovi utenti. **Gnome 3.14** ha il supporto per il multi-touch. In pratica, è possibile sfruttare completamente i comandi tattili per navigare nell'interfaccia desktop, usare le applicazioni, modificare la visualizzazione e molto altro ancora. Con applicazioni come **Evince** ed **Eye** di Gnome è possibile ingrandire e ruotare i documenti, nonché eseguire un pan sulle immagini. Le versioni future saranno caratterizzate da un numero sempre maggiore di applicazioni che supporteranno i comandi tattili. Gli sviluppatori, a questo proposito, possono utilizzare **GtkGestures** per aggiungere alle proprie applicazioni la maggior parte dei comandi multi-touch più comuni come **sfoglia**, **trascina**, **pinch**, **rotazione** e via dicendo. La versione 3.14 si concentra molto sul perfezionamento e il miglioramento del desktop, così da fornire un maggior numero di funzionalità. L'app **Meteo**, per esempio, è stata ridisegnata e ora offre un nuovo quadro di geolocalizzazione che mostra le previsioni del tempo. L'**Activities Overview** vanta nuove animazioni, nonché altrettante rinnovate transizioni per l'apertura, la chiusura e la riduzione delle finestre. Il tema visivo, compreso l'aspetto delle barre di avanzamento e delle finestre di dialogo, è stato completamente rivisitato. La calcolatrice e l'orologio sono poi stati aggiunti all'**Activity Overview**, così da essere utilizzati al volo per eseguire calcoli o controllare l'ora in qualsiasi città tramite la casella di ricerca.

Miglioramenti e nuove aggiunte

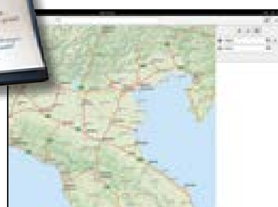
La maggior parte delle applicazioni ha qualcosa di nuovo da dare. Musica, per esempio, consente di creare le proprie playlist e offre anche una nuova funzione di ricerca che può fare da traino per la vostra collezione di musica locale o proveniente da fonti online come Jamendo. **Foto** ha il supporto per Google, in modo da configurare un account e accedere alle foto che sono state caricate tramite Picasa, Google+ o dispositivi Android. Non manca poi il supporto per Flickr e Facebook, anche se è necessario configurare preventivamente gli account dalle impostazioni di sistema. **Software** presenta un nuovo layout: le applicazioni hanno screenshot, descrizioni e classificazioni a stelle. In più, i componenti aggiuntivi per i programmi possono essere installati o rimossi direttamente all'interno dei software. Grazie alle capacità di virtualizzazione, è possibile eseguire delle istantanee di una macchina virtuale da caricare in caso di difficoltà, così da tornare a una configurazione funzionante. **Boxes**, la macchina virtuale di Gnome, consente ora di eseguire più sistemi in diverse finestre separate. Sul fronte della sicurezza, Gnome 3.14 offre la condivisione di rete: questo significa che memorizzerà quali reti richiedono l'abilitazione di WebDAV o VNC, così da non attivarli accidentalmente quando non servono. Tranne per qualche piccolo inconveniente, la 3.14 è una versione davvero ben progettata, che continua ad appianare i problemi riscontrati nella release 3. Certo, c'è ancora diverso lavoro da fare, ma il fatto che la strada imboccata sia quella giusta ci fa sperare. LXP



› **Gnome 3.14 segue il famoso copione di rilascio Open Source che fornisce una manciata di nuove caratteristiche e una moltitudine di miglioramenti per soddisfare vecchi e nuovi utenti**

In evidenza





Progetta il tragitto
Mappe pianifica i percorsi per viaggiare con ogni mezzo.



Touch multi-gesture
La 3.14 aggiunge il supporto per vari comandi touch.

Giudizio

Gnome 3.14
Produttore: The GNOME Project
Web: www.gnome.org
Licenza: Varie licenze Open Source

Caratteristiche	9
Prestazioni	8
Facilità d'uso	8
Documentazione	8

» *Provando questo aggiornamento è impossibile non mettere da parte le polemiche che hanno contraddistinto la versione 3.*

Il voto di Linux Pro 8

Kano Computer Kit

Si ritorna all'infanzia e si impara da capo a programmare grazie a un istruttore di Judo e a tanti blocchetti colorati

L'obiettivo di Kano è semplice: rendere la programmazione facile come il Lego. Alla fine del 2013 la sua campagna di finanziamento Kickstarter da 100.000 dollari ha raggiunto il suo obiettivo in 18 ore e ha raccolto in totale ben 1,5 milioni di dollari. Questo denaro ha finanziato la creazione e lo sviluppo dei componenti hardware, di una serie di opuscoli di istruzioni e del software Kano OS. In questo articolo ci occupiamo del pacchetto nel suo complesso.

Hardware

Kano utilizza il Raspberry Pi Model B e comprende accessori contrassegnati da specifici colori, come il dongle Wi-Fi verde, il cavo HDMI giallo e l'alimentatore rosso. I colori aiutano i bambini a imparare a montare il kit con l'aiuto degli opuscoli (sui quali torneremo tra poco). Un accessorio aggiuntivo è la tastiera wireless arancione con trackpad incorporato, che può essere utilizzata tramite il dongle incluso o via Bluetooth. Il kit è accompagnato da un robusto e solido involucro in plastica trasparente provvisto di altoparlante integrato di eccellente qualità.

Confezione e documentazione

Il kit Kano è distribuito in una confezione ben congegnata di ottima qualità. Comprende una chiusura magnetica, una trovata semplice che tuttavia la dice lunga sulla cura con cui il kit è stato realizzato. Le illustrazioni sulla custodia chiariscono ulteriormente il codice cromatico delle istruzioni di montaggio: è come se tutto quanto in questo pacchetto gridasse "gioca con me!". All'interno del coperchio ci sono i due libri di Kano: il primo fornisce un'introduzione a Kano e al montaggio del kit, il secondo si concentra sugli esercizi di programmazione integrati nel sistema operativo. I due libri procedono più o meno con lo stesso ritmo di un manuale di istruzioni Lego, il che permette ai bambini di 'imparare facendo', senza dover attendere che siano i genitori a montare il kit.

Software

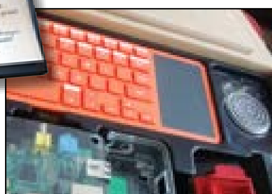
Kano OS si basa su Raspbian ma il team di Kano ci ha lavorato sopra in modo da dare vita a un sistema operativo più snello, il che, abbinato alla CPU overcloccata 'di listino', fa di Kano un sistema operativo decisamente scattante. Il team ha eliminato molti elementi dal kernel, come gvfs, zeitgeist e gnome-pty-helper; ma ha anche aggiunto alcuni miglioramenti, come 'copies-and-fills' che velocizza l'accesso alla RAM. Il team tiene a precisare che tutti gli eventuali miglioramenti verranno a loro volta integrati nel kernel di Raspbian. Su Github trovate l'elenco completo dei moduli kernel e dei software installati: l'indirizzo del progetto è **http://bit.ly/Kano_eVt**. Al primo avvio, Kano OS presenta una semplice esercitazione finalizzata a introdurre l'utente all'uso del kit e a suscitare il suo interesse: siamo invitati a seguire il coniglio nella tana e quindi a disinnescare una bomba digitando **startx**. L'interfaccia utente è fortemente stilizzata, con una miscela di colori vivaci e collegamenti rapidi ad applicazioni correlate all'obiettivo centrale del progetto, cioè la programmazione. Kano dà il meglio di sé nell'introdurre i bambini alla programmazione e lo fa per mezzo di una serie di esercizi e progetti che utilizzano varie applicazioni, tra cui Sonic Pi, Pong/Snake con l'impiego di Python e un favoloso sistema di programmazione in stile Scratch chiamato Kano Blocks, che viene utilizzato per programmare Minecraft. Il pacchetto Kano è ideale per bambini di ogni età, con la sua accattivante miscela di componenti facili da montare e progetti divertenti. Confezione, documentazione e componenti trasudano qualità e sono abbastanza robusti da sopportare i maltrattamenti da parte dei bambini. Il team di Kano ha creato un pacchetto eccellente ed entro breve affiancherà al kit un'intera gamma di accessori, a partire da una macchina fotografica basata su Pi. LXP



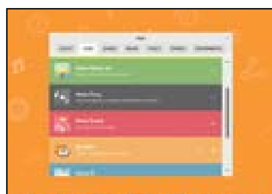
› Il kit con codice cromatico e la ricca interfaccia utente con le sue icone colorate e accattivanti stimolano l'interesse dei bambini e li aiutano a esplorare e a imparare al ritmo che preferiscono

In evidenza





Ottima confezione
La confezione è ben congegnata: il kit, per quanto sottilissimo, è decisamente accattivante.



Progetti a tutti i livelli
Kano comprende numerosi interessanti progetti, dai classici Pong e Snake a Sonic Pi e Minecraft.

Giudizio

Kano Computer Kit

Produttore: Kano
Web: **http://kano.me/**
Prezzo: 149,99 €

Funzioni	9
Prestazioni	9
Facilità d'uso	9
Qualità/prezzo	6

» *Un pacchetto di qualità zeppo di ottimi progetti e componenti in grado di arricchire l'apprendimento dei bambini di ogni età.*

Il voto di Linux Pro **8**

Lenovo W540

Una workstation 3K di fascia alta creata per Linux? Siamo decisamente entusiasti del ThinkPad W540 di Lenovo, malgrado il prezzo non proprio abbordabile

Gli utenti di Linux hanno ben poca scelta per quanto riguarda i portatili, scelta che si riduce ulteriormente quando si parla di workstation di alto livello. Provvista di certificato Ubuntu, il nuovo **ThinkPad W540** di Lenovo è tra i pochi modelli che soddisfano tutti i requisiti. Pur non avendo Ubuntu preinstallato, la workstation Lenovo comprende comunque il certificato completo Ubuntu 12.04. Con il suo display IPS da 15,6 pollici a risoluzione 2880 x 1620 (3K), il W540 è uno studio visivo mobile basato sulla piattaforma ThinkPad di Lenovo. Aggiungeteci il processore di fascia alta Intel Core i7-4800MQ a 2,7 GHz con scheda grafica Nvidia Quadro K2100M, ed ecco un dispositivo in grado di misurarsi con qualunque carico di lavoro, dalla semplice elaborazione di testi alla gestione di fogli elettronici da 1000 pagine, fino alla modellazione 3D. Di fronte a un prezzo non basso, tuttavia, è più che lecito domandarsi se il gioco vale la candela. Sul piano strutturale, il W540 non presenta alcuna caratteristica notevole. È infatti una fotocopia dei precedenti notebook ThinkPad. Quando è chiuso è un semplice rettangolo grigio privo di elementi decorativi, al di là di qualche bordo arrotondato. Ai suoi limiti stilistici, il W540 sopperisce però con un involucro in plastica rinforzato in fibra di carbonio che avvolge la struttura in magnesio. Questi due elementi permettono al portatile di sopravvivere a cadute e altri incidenti che danneggerebbero molti notebook simili.

Uno schermo d'eccezione

Grazie ai suoi lussuosi (e costosi) componenti, il W540 ha prodotto prestazioni impeccabili in tutte le applicazioni con cui lo abbiamo provato. Non abbiamo rilevato alcun rallentamento, né navigando in rete, né elaborando immagini, né visualizzando modelli di stampa 3D. Una nota a margine: il W540 non se la cava molto bene con i videogiochi ma il problema non è dovuto a carenza di potenza. La ragione, semmai, è che spesso gli sviluppatori di giochi non ottimizzano i loro prodotti per le schede grafiche da workstation, come quelle della linea Quadro. Il cavallo di battaglia del W540 è il suo spettacolare schermo 3K. A una risoluzione di 2800 x 1620, i pixel sono praticamente invisibili su questo display da 15,6". Per di più, lo schermo è luminoso e preciso nella riproduzione dei colori e chi necessita di una fedeltà cromatica di livello professionale può sempre calibrarlo con l'accessorio opzionale X-Rite Integrated Color Calibrator. Dopo la calibratura i colori sono risultati ancor più vividi e realistici di quelli del nostro MacBook Pro con display Retina utilizzato come pietra di paragone. Purtroppo, l'elevata risoluzione dello schermo implica un elevato consumo delle batterie. Nel nostro test sulle batterie il ThinkPad W540 ha resistito solo per tre ore e sette minuti. Con un utilizzo normale potete sperare di arrivare alle quattro ore e nove minuti, da suddividere in più di una decina di pagine Web aperte contemporaneamente, un po' di elaborazione di immagini e di visualizzazione di video 4K in streaming su YouTube. Anche utilizzandolo con moderazione e a luminosità ridotta, non siamo riusciti a superare le cinque ore e 17 minuti. Dunque vale o no la pena di acquistarlo? Oltre alla robusta qualità strutturale e alla praticità che abbiamo imparato ad aspettarci dai prodotti Lenovo, il W540 offre anche uno schermo assolutamente eccezionale. Testo e immagini presentano un contrasto eccellente, colori splendidi e una risoluzione straordinaria. Non abbiamo mai visto uno schermo simile su un portatile. Inoltre, il W540 dispone realmente di componenti abbastanza potenti da sopportare l'alta risoluzione di questo display 3K. Se vi occupate di progettazione grafica o fotografia, vi consigliamo decisamente di prendere in considerazione questo portatile. Gli unici nei sono rappresentati dall'autonomia della batteria, prevedibilmente limitata, dallo spazio su disco piuttosto modesto (500 GB HDD come standard o 256 GB SSD) e dal prezzo decisamente salato. LXP



› **Schermo spettacolare e certificazione Linux**

Scheda tecnica

- **CPU:** Intel Core i7-4800MQ 2,7 GHz
- **GPU:** Nvidia Quadro K2100M 2 GB; Intel HD Graphics 4600
- **RAM:** 8 GB DDR3
- **Schermo:** IPS da 15,5 pollici, 2880 x 1620 (3K)
- **SSD:** 256 GB
- **Ottica:** DVD+/-R/RW DL 8x
- **Porte:** VGA, 2 USB 3.0, Express Card, lettore schede 4-in-1, ingresso DC In, Thunderbolt, mini DisplayPort
- **Wi-Fi:** 802.11ac, Bluetooth 4.0
- **Dimensioni:** 50 x 31 x 7 cm
- **Peso:** 2,53 kg

Giudizio

Lenovo ThinkPad W540

Sviluppatore: Lenovo
Web: www.lenovo.com/it
Prezzo: 1.907,58 €

Funzioni	9
Prestazioni	10
Facilità d'uso	9
Qualità/prezzo	7

» *Ha uno schermo superbo ma è caro. Le alternative sono molte: alcune offrono perfino schermi a risoluzione più elevata a meno.*

Il voto di Linux Pro **8**

ReadyNAS 716

Abbiamo indossato il nostro casco protettivo e ci siamo apprestati a provare questo colosso 10GbE pubblicizzato da Netgear come il NAS più veloce del mondo

Presentato da Netgear come il NAS desktop più veloce del mondo, il **ReadyNAS 716** a sei alloggiamenti si guadagna questo riconoscimento in virtù del suo processore quad-core Intel Ivy Bridge Xeon E3 e dei suoi 16 GB di RAM. Non è tutto: oltre che su una coppia di porte Gigabit, potrete contare anche su due interfacce 10GbE di listino. Queste specifiche permettono al ReadyNAS di competere su un piano di parità con un server di fascia media. Anche il prezzo, d'altronde, è paragonabile: si parte da 3.000 € circa, dischi esclusi, che probabilmente eccedono il potere d'acquisto di molte piccole imprese. L'investimento potrebbe essere tuttavia giustificato per le piccole e medie imprese che aspirano a un sistema di archiviazione veloce e di livello professionale in formato desktop. Il dispositivo comprende alloggiamenti per dischi SATA da 3,5 o 2,5": è sufficiente inserirli nei sei alloggiamenti disposti verticalmente dietro il portello anteriore. Un piccolo gancio presente su ciascun alloggiamento impedisce la rimozione accidentale ma non sono presenti blocchi fisici, né negli alloggiamenti né sul portello.
Le quote predefinite vengono create in fase di installazione ma è possibile configurarle ulteriormente (insieme ai volumi iSCSI) mediante i consueti comandi durante la procedura di autenticazione e di accesso da parte dell'utente. La più recente implementazione del firmware ReadyOS si caratterizza inoltre per il passaggio dal file system EXT4 a BTRFS, che tra l'altro introduce il supporto per le istantanee a livello di blocco. In più, è possibile creare un numero illimitato di istantanee dei volumi iSCSI e delle quote di network; semplici strumenti permettono di controllare la procedura e di ripristinare il sistema a un'istantanea precedente mediante un display cronologico.
Per i nostri test abbiamo utilizzato un Netgear ProSafe XS708E. Avvalendoci di un server Fujitsu per effettuare l'archiviazione sul ReadyNAS siamo riusciti a raggiungere una spettacolare velocità di 420 MB al secondo nella scrittura di file di grandi dimensioni con una connessione 10GbE, con punte di 480 MB al secondo in lettura. Durante la copia di file più piccoli, tuttavia, la velocità è risultata inferiore; abbiamo rilevato un rallentamento anche dopo la sostituzione delle unità SSD con dischi magnetici. Alcuni utenti, ovviamente, si serviranno soltanto della rete Gibabit; e con le sue porte Gigabit il ReadyNAS 716 è risultato in grado di sfruttare appieno la larghezza della banda disponibile. In pratica, prevediamo che questi utenti si serviranno di tutte e quattro le porte sfruttando le varie opzioni di configurazione dei trunk disponibili nel firmware Netgear, oppure servendosi della segmentazione VLAN per passare eventualmente in seguito a 10GbE.
Per quanto riguarda l'estetica, il ReadyNAS 716 è uno scatolone nero che può piacere o meno; lo stesso vale per il pannello di stato sensibile al tocco presente sul portello. A noi è piaciuto, anche se una volta avviato il sistema è difficile che il display incorporato vi serva a qualcosa, se non a fare colpo sui visitatori.
L'interfaccia di gestione Web è l'ambito nel quale trascorrerete la maggior parte del tempo in fase di gestione. Noi l'abbiamo trovata datata ma molto piacevole da usare e facile da imparare, con strumenti molto semplici che permettono per esempio di gestire i volumi e di collegare tra loro gli adattatori di rete. Sono inoltre disponibili numerosi strumenti di monitoraggio; ma sono le prestazioni l'ambito in cui il ReadyNAS 716 eccelle veramente e sotto questo aspetto il dispositivo Netgear è praticamente imbattibile, a meno di optare per sistemi montati su rack. Naturalmente saranno molte le aziende sprovviste di spazio per i rack, che troveranno più pratico utilizzare il ReadyNAS. Tuttavia, l'assenza di una chiusura per il sistema di archiviazione sembra un mezzo passo falso in questo contesto e il dispositivo in sé è un po' ingombrante, anche tenendo conto dei suoi sei alloggiamenti. Inoltre, con un prezzo del genere non sarebbe stata fuori posto un'opzione di alimentazione ridondante per desktop o per l'utilizzo in rack.
Le prestazioni rappresentano la principale attrattiva del ReadyNAS 716, in particolare in relazione alle interfacce 10GbE presenti come standard, una novità assoluta per un NAS desktop. Il dispositivo ha molto da offrire anche in termini di capacità (è possibile espanderlo ben oltre i 28 TB base) e dispone di opzioni di backup e creazione di istantanee di livello professionale. L'unico neo è il prezzo, che lo colloca nell'arena dei server di fascia media e dei dispositivi di archiviazione montati su rack, un ambito in cui i suoi vantaggi appaiono meno netti. LXP



› **Il ReadyNAS 716: un'unità NAS desktop ultra-veloce... ma costosa**

In evidenza





Interfaccia Web

L'interfaccia di gestione è nitida ma datata. Avremmo preferito qualcosa di più funzionale.



L'armadietto dei liquori

OK, le bottiglie di whisky non ci stanno; i dischi da 3,5 e 2,5 pollici sì... ma non c'è alcuna chiusura, perciò chiunque può rimuoverli!

Giudizio

ReadyNAS 716

Produttore: Netgear
Web: **www.netgear.it**
Prezzo: 3.000 € ca (senza dischi)

Funzioni	10
Prestazioni	10
Facilità d'uso	8
Qualità/prezzo	7

» *È il NAS desktop più veloce del mondo; ma il suo prezzo lo colloca nell'arena dei server e dei dispositivi di archiviazione montati su rack.*

Il voto di Linux Pro 9

Tox

Perennemente in cerca di modi per mantenere riservate le nostre conversazioni, abbiamo provato un client di messaggeria istantanea criptato deciso a detronizzare Skype

Il client **Tox Instant Messaging** è il risultato di un'accesa conversazione su 4chan riguardo alla necessità di un sostituto di Skype che fosse Open Source, attento alla sicurezza e decentrato. Tox soddisfa tutti e tre i requisiti. Due sono le tecnologie vitali per Tox: la cifratura e il P2P. Questo client di messaggeria istantanea poggia sulla stessa tecnologia utilizzata da BitTorrent per creare connessioni dirette tra gli utenti, eliminando la necessità di un nodo centrale che potrebbe essere compromesso o eliminato. Il programma si serve dei cosiddetti Tox ID, chiavi pubbliche di peer usate in luogo dell'account utente e consente inoltre un maggiore anonimato. Inoltre, tutte le chat sono criptate mediante la libreria di cifratura NaCl. Non essendovi alcun server centrale, gli utenti devono semplicemente avviare il client Tox e aggiungervi i loro amici, senza doversi iscrivere a un servizio o configurare un account. Ogni utente di una rete Tox è rappresentato da una stringa di byte, che costituisce il suo Tox ID. Tox si serve di una tabella di hash distribuita (DHT) in stile torrent, che permette ai peer di trovare l'indirizzo IP di altri peer utilizzando il proprio Tox ID. Dopo aver ottenuto l'indirizzo IP, gli utenti possono stabilire tra loro una connessione sicura. Il sito Web del progetto offre versioni del client per i più diffusi sistemi operativi, tra cui Linux, Windows e OS X, nonché per i sistemi operativi mobili Android e iOS. Il progetto dispone inoltre di repository per le distribuzioni basate su APT e RPM, che vi permetteranno di far funzionare Tox sulle principali distribuzioni desktop senza faticare troppo. Dopo aver aggiunto il repository potrete installare uno dei client supportati: uTox, qTox, Toxic e Venom. Sono tutti in fasi di sviluppo diverse; stando ai nostri test, comunque, tutti i client per desktop funzionano meglio di quelli per sistemi mobili e offrono più funzioni. Le modalità d'uso sono sostanzialmente le stesse per tutti i client. Avviate il programma, create un nickname che genera a sua volta un Tox ID che potrete quindi comunicare agli amici. Quando gli amici vi inseriranno nel proprio elenco riceverete una notifica, che dovrete accettare perché sia possibile stabilire la connessione. Se l'amico è vicino a voi e utilizza un client per sistemi mobili, potrà aggiungervi anche scansionando un codice QR del vostro Tox ID.

Comunicare in Tox

Abbiamo provato i client µTox e Venom su desktop insieme al client Antox su Android e li abbiamo trovati semplicissimi da capire e da utilizzare.



› **Potete usare Tox anche con Pidgin, utilizzando il plug-in tox-prpl disponibile su http://tox.dhs.org**

I messaggi di testo istantanei funzionano su tutti i sistemi operativi supportati, per cui un utente di Venom che utilizza Ubuntu può inviare messaggi di testo a un utente di µTox che utilizza Windows o a un utente di Antox che utilizza Android. Gli utenti di desktop che si servono dei tre client possono scambiarsi chiamate audio e video e non abbiamo rilevato ritardi o distorsioni nel nostro ambiente di prova. Allo stesso modo, tutti e tre i client possono anche effettuare trasferimenti di file: abbiamo inviato di tutto, da piccole immagini a grossi file ISO e tutto quanto è arrivato all'altro utente senza problemi. Ci sono tuttavia un paio di elementi su cui gli sviluppatori stanno ancora lavorando attivamente. Il più importante è la possibilità di effettuare chiamate audio e video mediante il cliente per Android, Antox, cosa attualmente non possibile. Inoltre, per il momento è possibile scambiare messaggi di testo solo all'interno di una chat di gruppo con più partecipanti. Di solito non recensiamo software di cui non è ancora disponibile una versione stabile ma Tox fa eccezione per due motivi. Primo, è estremamente pratico, almeno su desktop. Secondo, è più di un client di messaggeria istantanea: di fatto è un protocollo per la trasmissione criptata di dati peer-to-peer. Infatti alcuni sviluppatori stanno utilizzando questo protocollo per sviluppare client sicuri di posta elettronica P2P e applicazioni sicure per la sincronizzazione dei file. In più, per mettere ancor più a suo agio la comunità degli utenti, gli sviluppatori hanno intenzione di rivolgersi a un'impresa specializzata nella sicurezza perché controlli il codice quando il progetto raggiungerà un maggiore livello di stabilità. LXP

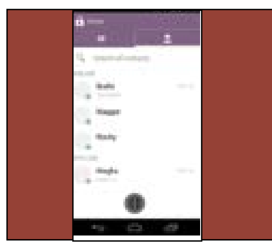


In evidenza



Comunicazioni sicure

Il sistema, creato con un occhio di riguardo per la sicurezza, utilizza gli algoritmi di cifratura della libreria NaCl.



Per tutti i sistemi

Sono disponibili client per sistemi fissi e mobili tra cui Linux, Windows, OS X, FreeBSD, Android e iOS.

Giudizio

Tox

Sviluppatore: The Tox Foundation
Web: tox.im
Licenza: GNU GPL v3

Funzioni	7
Prestazioni	8
Facilità d'uso	7
Documentazione	6

» *Benché ancora nelle fasi iniziali dello sviluppo, è molto pratico ed efficiente, sebbene per ora sia disponibile solo per desktop.*

Il voto di Linux Pro **7**

Untangle NG 11

Dovete creare una rete locale? Bene, abbiamo puntato l'attenzione su una distro che adotta un approccio tutto particolare. Peccato non faccia niente per differirsi dai principali concorrenti

La versione 11 della distro Debian-based **Untangle Next Generation Firewall** è disponibile in due edizioni: **Free** e **Complete**. La prima è un'immagine ISO installabile. A differenza di una distro desktop, non è disponibile alcun ambiente live. Dovrete quindi procedere all'installazione fin da subito. La distro funziona come server gateway, mentre il suo installer Debian modificato è stato progettato per prendere in consegna l'intero disco. Tutto quello che dovete fare è puntare alla partizione giusta. Una volta installato, il computer si riavvia con una procedura di configurazione guidata. Viene chiesto di impostare una password di amministrazione, quindi selezionare e configurare due profili di rete: il primo per la connessione a Internet e il secondo per il collegamento alla rete locale. Una volta terminata l'installazione, Untangle vi chiederà di creare un account per la configurazione del server. La prima cosa da fare quando si accede al server gateway di Untangle è installare le applicazioni desiderate.
Le "recommended" offrono una dozzina di programmi e servizi che possono essere installati con un solo click. Troviamo filtri Web, antivirus, spam blocker, programmi per il controllo della larghezza di banda e delle app, WAN balancer, un firewall e molto altro ancora.

Facile e intuitivo

Quasi tutte le applicazioni sono preconfigurate e funzionano automaticamente dopo l'installazione. Altrimenti è possibile configurarle facendo click sul relativo pulsante **Impostazioni**. Per qualsiasi bisogno, è sufficiente visitare la pagina Wiki di Untangle, dove potrete trovare qualsiasi informazione di cui abbiate bisogno, tra cui tutorial video. Dopo aver configurato un'applicazione, basta premere il rispettivo pulsante per attivare il servizio. Dalla console di configurazione avrete anche la possibilità di controllare il traffico che passa attraverso il server gateway. Ogni applicazione, inoltre, vi illustrerà le varie statistiche per le proprie attività di rete. Il servizio per il blocco della pubblicità, per esempio, vi informerà di quanti annunci ha bloccato, l'antivirus mostrerà quanti file sono stati scansionati e il filtro Web quante pagine sono state bloccate. Purtroppo, la maggior parte di questi servizi è a pagamento e con la versione Free di Untangle possiamo saggiarne le funzioni per soli 14 giorni. È possibile acquistare una licenza per ogni singola applicazione. Il prezzo varia a seconda del programma e dipende dal numero di dispositivi di rete e dalla durata della sottoscrizione. È però possibile acquistare il pacchetto completo che include tutte le applicazioni a pagamento. Ancora una volta il prezzo varia in base ai due fattori che abbiamo poco sopra specificato. A titolo d'esempio, un abbonamento annuale al pacchetto completo che copre fino a dieci dispositivi costa circa 340 euro. Questo include anche servizi aggiuntivi come il branding personalizzato e l'assistenza.
Ci domandiamo se ne valga effettivamente la pena.
Se consideriamo che Zentyal e ClearOS sono due tra le più agguerrite distro concorrenti e vengono messe a disposizione gratuitamente, è lecito chiedersi se sborsare quasi 350 euro per avere gli stessi servizi sia davvero un affare. Entrambe le distro hanno perfezionato molto l'interfaccia delle applicazioni e dei rispettivi moduli supportati, in più offrono un processo di configurazione semplice e veloce. Solo Zentyal mette a disposizione un plus di servizi accessori dietro pagamento di una sottoscrizione. Tuttavia, Untangle, Zentyal e ClearOS hanno le stesse funzionalità, ma dal momento che ognuna è corredata da applicazioni diverse, non ci sentiamo di escludere che una distro possa comportarsi meglio dell'altra in talune circostanze. Detto questo, Untangle non fa niente per distinguersi da Zentyal che continua a essere la nostra scelta principale. LXP

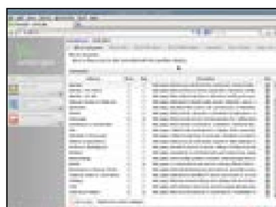


› **Le applicazioni della versione Free di Untangle sono disponibili per un periodo di prova di soli 14 giorni, trascorsi i quali, se volete continuare a utilizzarle, dovete sottoscrivere una licenza a pagamento**

In evidenza



Onnicomprensiva
A livello di applicazioni, non manca proprio niente.



Facile da gestire
Per l'amministrazione potete sfruttare il front-end grafico.

Giudizio

Untangle NG Firewall Free 11

Produttore: Untangle Inc.
Web: **www.untangle.com**
Licenza: Varie

Caratteristiche	5
Prestazioni	8
Facilità d'uso	7
Documentazione	9

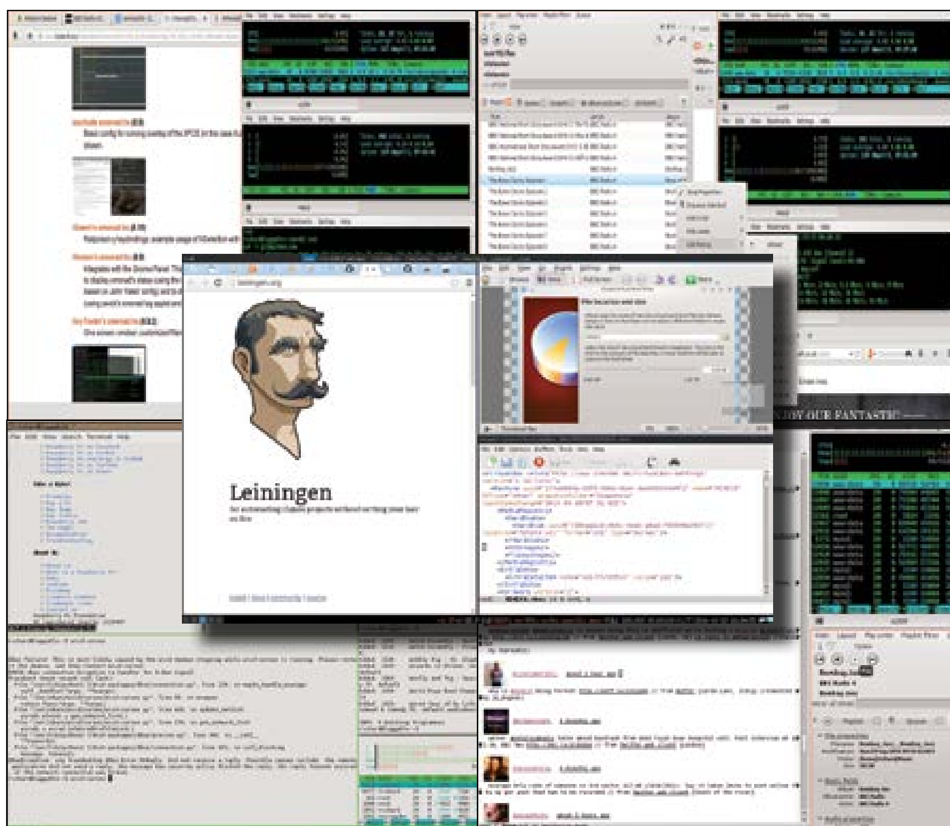
» *La versione gratuita è poco più di una demo della release a pagamento. Rispetto alle distro gratuite c'è davvero poco da guardare.*

Il voto di Linux Pro **7**

Ogni mese mettiamo a confronto prodotti e programmi per farvi scegliere al meglio!

Windows Manager

Le moderne distro Linux sono ricche di widget e programmi che danno un grande risalto alla parte grafica. A noi, però, piace fare le cose con un approccio minimalista. Vediamo quali sono gli strumenti migliori



Modalità del test

In questo confronto non andremo a valutare la filosofia minimalista delle interfacce più comuni. Ci siamo invece concentrati sulla loro produttività e quindi sull'approccio pratico, la loro semplicità d'uso e le potenzialità che le contraddistinguono. Per migliorare le prestazioni di un PC piuttosto obsoleto dobbiamo fare a meno di molti strumenti grafici, dal gestore della rete Wi-Fi al monitoraggio delle risorse. Per non rimanere senza queste funzioni esistono comunque alternative molto più leggere che consentono di sostituire le barre di stato e i menu. Eseguendo queste modifiche, abbiamo valutato l'impatto che hanno avuto sull'operatività del sistema. La scelta dei cinque Windows Manager (WM) non è stata facile e prima di decidere ne abbiamo provati parecchi, sia in configurazione su PC portatili sia su multi-monitor.

La nostra selezione

- XMonad
- StumpWM
- i3
- TMux
- Emacs

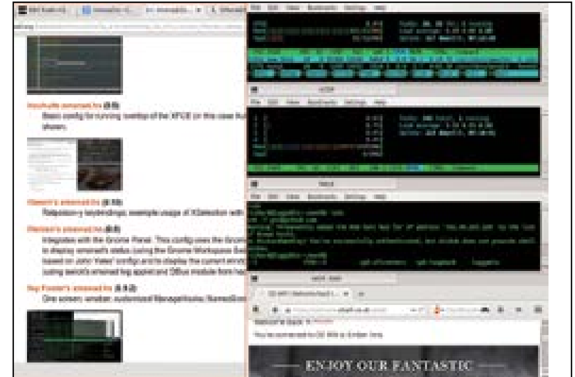
Le distro che usano Unity o KDE sono delle meraviglie moderne. Ricche di menu, widget e altri accessori che nella maggior parte dei casi rendono la vita più semplice, oltre che più colorata e allegra. Tuttavia, configurazioni del genere possono non essere sempre la migliore soluzione per tutti. Basta pensare alle persone che utilizzano ancora un computer piuttosto obsoleto e poco propenso a gestire gli orpelli grafici. Oppure a tutti coloro che usano il PC principalmente per scrivere, programmare o quant'altro non richieda alcuna attenzione grafica. In questo caso, la maggior parte dei desktop moderni non sono altro che grandi distrazioni di massa. Per fortuna esistono Windows Manager (WM) molto più minimalisti e che hanno dalla loro un minimo consumo di risorse. Di tutti i WM in circolazione, abbiamo deciso di valutarne tre, più due strumenti a riga di comando che possono essere molto utili in ambito produttivo. Stiamo parlando del terminale multiplexer **TMux** e di **Emacs**, l'editor di testo cresciuto talmente tanto da diventare addirittura un vero e proprio sistema operativo.

> **"Gli effetti grafici della maggior parte dei desktop non apportano alcunché alla produttività"**

Che tipo di curva di apprendimento

Quanto ci vuole per imparare a usarli?

Passare da un ambiente ricco di grafica a un'interfaccia minimalista, con finestre e schede completamente diverse, può essere scoraggiante. **XMonad**, per esempio, una volta installato vi lascia davanti a una schermata vuota, senza neppure il cursore lampeggiante. Per avviare il terminale è necessario premere **Alt-Invio**, mentre **Alt-P** permette di inserire un comando per aprire un programma attraverso **dmenu**. Premendo da **Alt-1** ad **Alt-10** potrete aprire dieci desktop con cui divertirvi. i3 sfrutta "Mod4" che applica alla tastiera un sistema piuttosto ridondante in chiave "Windows". **Mod-D** apre il menu come **Alt-P** in XMonad, mentre **Mod-Enter** lancia il vostro xterm di default. Se non siete nuovi all'uso di WM minimalisti, probabilmente avrete già sentito parlare di Ratpoison, di cui **StumpWM** è una riscrittura in Lisp. Imitando la gestione della tastiera di GNU Screen, StumpWM usa tasti di scelta rapida preceduti da **Ctrl-t** anziché utilizzare una chiave Mod. StumpWM è il WM che grazie alla sua documentazione vi guida nel modo più naturale possibile verso l'apertura di editor, terminale, browser e via dicendo. In più consente di passare dall'uno all'altro programma con facilità, suddividendo lo schermo in più finestre. **Ctrl-t h** richiama i comandi di base, mentre **Ctrl-t !** permette di lanciare qualsiasi programma. Emacs, contrariamente alla sua reputazione, non è da difficile da utilizzare. Si possono sfruttare innumerevoli trucchi e in più si può accedere a molte funzionalità tramite i menu. Tuttavia, per usare al meglio Emacs e ottimizzare la gestione dei flussi di lavori ci vorrà un po' di tempo. Il modo migliore per imparare è immergersi per un paio di giorni nella stesura di un progetto (la creazione di uno script di posta elettronica o di qualsiasi altra applicazione). Una volta che vi sentite a vostro agio, allora potete iniziare a espandere l'uso di Emacs, utilizzando sempre nuove funzioni. Per capire a pieno TMux, infine, bisognerebbe scordarsi l'uso di qualsiasi applicazione che non sia a riga di comando. Se da una parte questo può essere un bene, soprattutto per chi usa frequentemente sessioni SSH, dall'altra non è l'ideale per tutte le attività. Le basi non sono comunque difficili da imparare. Servono solo un paio d'ore di buona volontà.



› **XMonad è tra i WM più immediati da utilizzare fin dal primo istante. Il file di configurazione, però, non è altrettanto intuitivo**

Verdetto

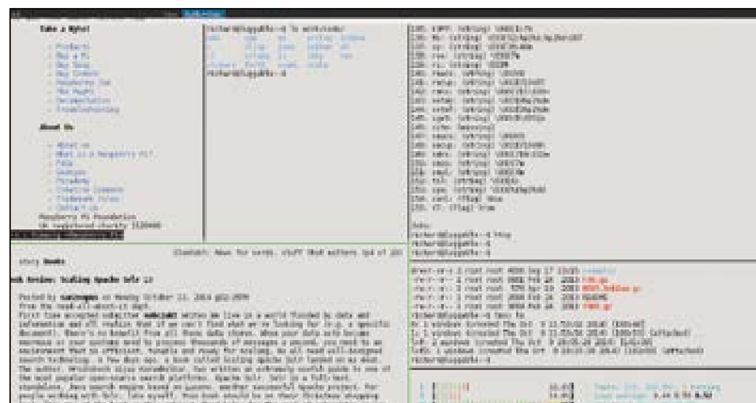
XMonad ★★★★★
StumpWM ★★★★☆
i3 ★★★★☆
TMux ★★★☆☆
Emacs ★★★★☆

» *Tutti i nostri WM sono piuttosto semplici da usare fin dal primo momento.*

Finestre e desktop

Dalla gestione delle finestre al multi-monitor

Affiancare le finestre è un modo di lavorare accattivante e minimalista. Per impostazione predefinita, una finestra aperta riempie l'intero schermo fino a quando non lo deve condividere con un altro programma. Aggiungere desktop virtuali permette di utilizzare lo spazio a disposizione in modo molto più efficiente. Per configurazioni multi-monitor, i nostri WM utilizzano Xinerama a cui hanno applicato qualche ritocco. In questo caso, l'aggiunta di una riga a xmonad.hs consente di sfruttare il comando **M-Esc** per cambiare le schermate. **i3**, pur non funzionando male in questo contesto, non è il migliore. KSnapshot si è comportato bene sul desktop, così come l'apertura di Gimp, ma i risultati non sono stati ineccepibili. StumpWM, pur essendo piuttosto povero di programmi multi-finestra come Gimp, riesce a destreggiarsi bene tra diverse applicazioni aperte contemporaneamente. XMonad sfrutta invece diversi algoritmi selezionabili per l'uso delle tile al di fuori della finestra principale. Sotto questo profilo, StumpWM e i3 generano tile apposite nel momento in cui si divide il desktop in orizzontale o verticale. Lo stesso fanno Emacs e TMux. Quest'ultimo, dividendo la finestra in riquadri separati, consente un uso piuttosto facile dell'ambiente di lavoro, anche se ci vuole un po' di tempo per prendere confidenza con gli spostamenti tra l'uno e l'altro.



› **TMux consente di dividere lo schermo in quante porzioni volete**

Verdetto

XMonad ★★★★☆
StumpWM ★★★★★
i3 ★★★★☆
TMux ★★★★★
Emacs ★★★★☆

» *StumpWM e TMux sono i migliori sotto questo punto di vista. Suddividono lo schermo molto bene e permettono di lavorare su più sessioni.*

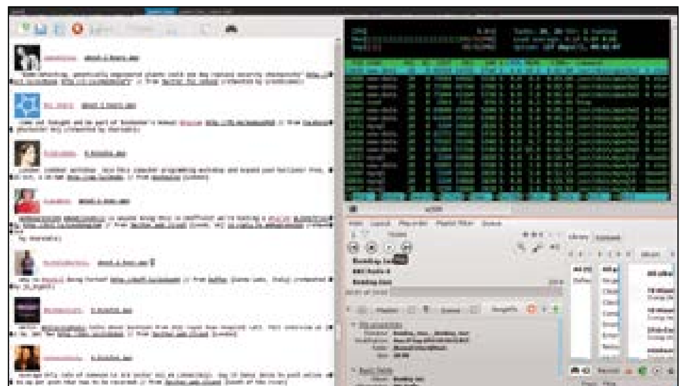
Potenza produttiva

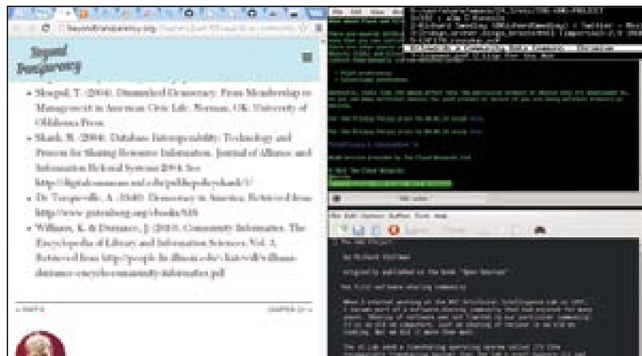
Velocità ed efficienza: è tutto sotto controllo

In un ambiente desktop, dove il vero punto di forza è il controllo di ciò che avete davanti, è importante che l'interfaccia compia a dovere ciò per cui è stata progettata. Dalla gestione delle finestre agli spazi di lavoro (desktop virtuali), fino al lancio delle applicazioni, un WM produttivo deve fare ciò per cui è stato progettato senza restituire alcun problema. Tuttavia è importante fare una precisazione: il nostro giudizio, in questo senso, è molto personale. Abbiamo dedicato diversi giorni a provare ciascun WM, sforzandoci di adattarlo al nostro flusso di lavoro. Tuttavia, quello che può andare bene per noi può non essere ideale per tutti. Ognuno deve sentirsi a proprio agio con l'ambiente che utilizza e questa sensazione è profondamente soggettiva. Alcuni preferiscono TMux, altri Emacs. In altre parole, si tratta solo di scegliere qual è il miglior WM che fa al caso vostro.

XMonad ★★★★☆

Il cavallo di battaglia di XMonad è la sua capacità di mettervi subito a vostro agio. Gli spazi di lavoro sono già assegnati e l'ambiente è perfettamente organizzato. Abbiamo subito trovato **Alt-3** per aprire il nostro editor. Siamo poi stati in grado di ridimensionare le tile senza alcuna difficoltà. Uno dei problemi che però abbiamo notato con **Alt-<Numero>** è nel cambiamento dello spazio di lavoro, che non permette di navigare tra le schede di Firefox o di un terminale Gnome. Per farlo, abbiamo dovuto mettere mano al file di configurazione. È bene precisare che per gestire **xmonad.hs** (il file di configurazione) è necessario conoscere una base di Haskell. Ciononostante è sufficiente prendere confidenza con i principali comandi, per riuscire nell'intento.





StumpWM ★★★★☆

StumpWM ha il pregio di mantenere tutto a portata di mano: le finestre e le schede sono facilmente controllabili dalla tastiera. Basta usare **Ctrl-t w** per avere una lista delle finestre tra cui passare e **Ctrl-t o** per muoversi tra le schede presenti nel programma correntemente utilizzato. Con una semplice divisione dello schermo (**Ctrl-s**/**Ctrl-S**) ci si sente subito a casa propria. Tra gli aspetti migliori di questo WM c'è l'ampia possibilità di configurazione. Quando inizierete a prendere confidenza con la personalizzazione, entrerete nel vivo di StumpWM. Con i tasti di scelta rapida, oltre che aprire le applicazioni, potrete gestire anche diversi menu. I comandi sono poi modificabili, così da essere resi ancora più intuitivi: **Ctrl-t**, per esempio, può diventare **AltGr-**.

Controlli da tastiera

Come ci si trova a interfacciarsi con l'ambiente senza mouse?

Tmux, senza le dipendenze X, non è nemmeno in grado di supportare l'interazione con il mouse. Se quindi siete alla ricerca di un ambiente che non vi faccia alzare le mani dalla tastiera, allora questo è sicuramente quello che fa per voi. Grazie poi alla sua ampia configurabilità, potrete cucirvi addosso un sistema che faccia esattamente quello che volete. Lo stesso concetto vale per i3 che oltretutto fa un ottimo lavoro nel ridimensionamento delle finestre con una modalità appositamente dedicata (**Mod-r** per entrare, **Esc** per uscire). StumpWM, come si addice ai WM che si ispirano a GNU Screen, ha bisogno di qualche input proveniente dal mouse. Rispetto alle altre soluzioni, è necessario perdere un po' di tempo a imparare. Tuttavia, nonostante la configurazione basata su Lisp, in quanto a potenza nella produttività possiamo avvicinarlo a TMux. XMonad e i3 funzionano entrambi molto bene, anche se nella maggior pare dei casi dovrete mettere mano al file di configurazione piuttosto che ai comandi da tastiera. Emacs, se utilizzato con la propria GUI, pur avendo diverse opzioni accessibili dai menu tramite il mouse, può essere sfruttato interamente da riga di comando. Pertanto, anche in questo caso, non avrete bisogno di alzare le mani dalla tastiera. Emacs, tramite Emacspeak, può essere utilizzato anche dai non vedenti, fornendo un desktop audio completo.

Verdetto

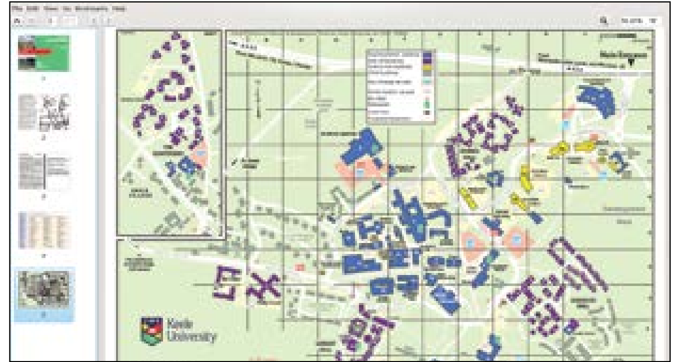
XMonad ★★★★☆
StumpWM ★★★★★
i3 ★★★★☆
TMux ★★★★★
Emacs ★★★★★

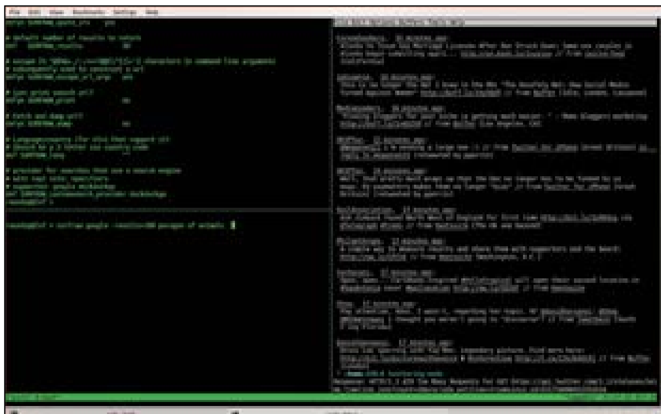
» Tra tutti i WM, sotto questo aspetto, Emacs è sicuramente il migliore. Può essere utilizzato anche dai non vedenti.

i3 ★★★★★

i3 è uno dei WM preferiti dagli amministratori di sistema. Si presenta con una grande quantità di riquadri e schede informative. È molto semplice da configurare, soprattutto dopo aver dato una lettura alla documentazione disponibile nelle varie risorse online. Potete impostare diverse aree di lavoro per passare automaticamente da un monitor all'altro, oppure impostare solamente il comportamento di una specifica finestra.

Ogni tasto di scelta rapida può essere modificato: niente vieta di usare **Alt** o qualsiasi altro pulsante. Come XMonad, i3 lavora bene con diverse istanze di Gnome, come il daemon keyron o il gestore di rete. La barra di stato di i3 mette poi a disposizione una serie di informazioni sullo stato del sistema, in modo simile a quanto riesce a fare Conky.



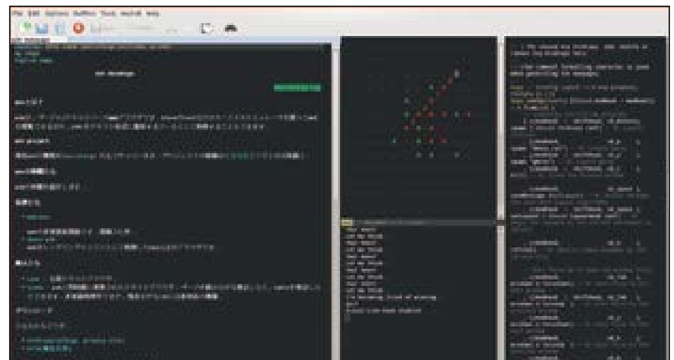


TMux ★★★★★

Se i WM che sfruttano le tile riescono in parte a incrementare la produttività eliminando le distrazioni, TMux fa molto di più. Le sue sessioni sono estremamente modulari e possono essere gestite nella più totale libertà. Potete per esempio tenere separate le applicazioni che utilizzate nel tempo libero da quelle che invece usate durante il lavoro. TMux, inoltre, è estremamente modulare: si può partire da una sessione minimale, per poi aggiungere tutto ciò che serve mano a mano che si presenta la necessità. Sempre seguendo questa filosofia, TMux mette a disposizione una vasta gamma di componenti aggiuntivi che possono servire per i più disparati scopi. A questo proposito, alla pagina **github.com/tmux-plugins**, ne trovate una buona scelta, con in più molte informazioni e tutorial utili all'installazione e alla configurazione di ciascuno.

Emacs ★★★★★

In Emacs, qualsiasi cosa, dai terminali ai file di debug, è aperto. Dal punto di vista desktop, volendo fare un paragone più accentuato con i3, questo non significa solo passare da un elemento all'altro utilizzando la tastiera ma anche gestire perfettamente qualsiasi altro processo. Se la vostra tastiera non è suddivisa in modo ergonomico, potete modificare i due tasti di controllo **Alt**, così che possano essere raggiunti con il pollice. Questo vi permetterà di utilizzare l'ambiente in modo molto più comodo e versatile. Emacs è poi uno dei WM più potenti dal punto di vista dell'integrazione. È possibile aggiungere di tutto: da un client Twitter fino a PIM. In altre parole, Emacs, prima ancora di essere un grande IDE, è diventato nel tempo uno dei migliori ambienti desktop attualmente in circolazione.



Configurazione

Non c'è nessun wizard qui, ma solo file di configurazione

L'installazione dovrebbe permettervi di scegliere il WM direttamente dal menu di login. Tuttavia, StumpWM non ha creato alcun file **stumpwm.desktop** in **/usr/share/xsessions/** sul nostro notebook con Lubuntu e quindi ci siamo basati su **wmii.desktop**. TMux è un semplice file che permette la riconfigurazione di ogni comando, tra cui il prefisso **Ctrl-b** che viene anteposto a tutte le istruzioni di TMux. La pagina **man** riporta i comandi da tastiera più importanti che ci permettono di prendere subito il controllo del sistema. i3, invece, durante l'installazione permette di scegliere una serie di tasti predefiniti. Il file di configurazione è un semplice testo molto ben commentato che potete trovare in **~/.i3**. Se volete approfondire al massimo le modalità d'uso di questo WM, allora il punto da dove iniziare è sicuramente la pagina man. Il file di configurazione di XMonad è scritto in Haskell e non è per niente intuitivo. Quello di StumpWM è basato su Lisp e, al pari di quello di XMonad, rappresenta un grosso ostacolo per tutti coloro che non conoscono a dovere il linguaggio di programmazione in questione. Emacs, per il proprio file di configurazione, utilizza Elips. Pur non essendo particolarmente difficile, ha una curva di apprendimento abbastanza ripida che potrebbe ostacolare i principianti.

Verdetto

XMonad ★★☆☆☆

StumpWM ★★★☆☆

i3 ★★★★★

TMux ★★★★☆

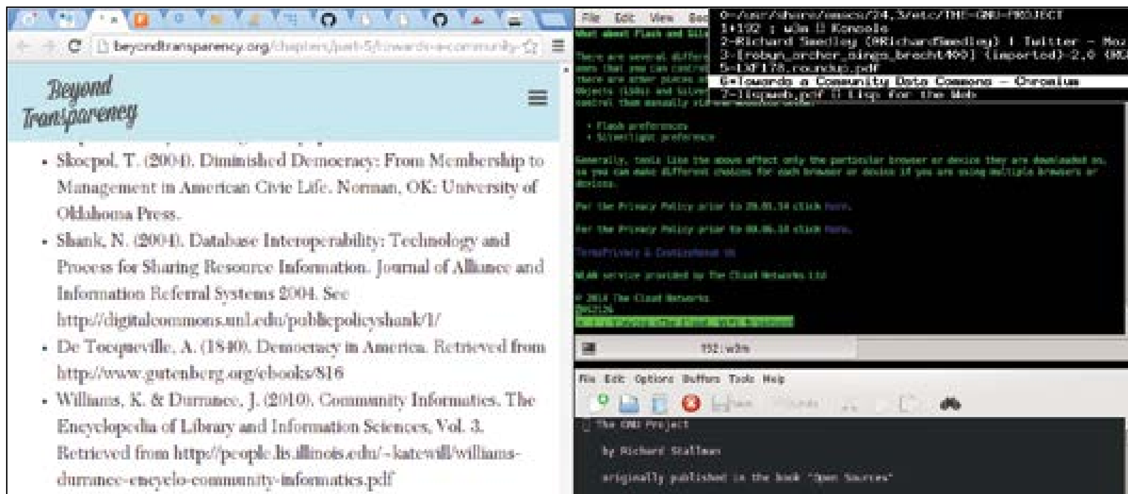
Emacs ★★★★★

» È vero che il file di configurazione di Emacs non è tra i più facili da gestire, ma dopotutto è lo scotto da pagare per la sua potenza.

Gestione dello spazio

Meno fronzoli equivale a massimo spazio di lavoro

Sotto il profilo che andiamo ad analizzare qui, XMonad e StumpWM ottengono entrambi un buon punteggio. i3, invece, non è altrettanto valido nella gestione delle schermate. Questo anche se la barra presente nella parte inferiore dello schermo è dotata di svariati comandi utili al passaggio da un desktop all'altro, al controllo dello spazio su disco e a quello dello stato della batteria e via dicendo. La barra sopra le finestre, tuttavia, è a nostro parere uno spreco di spazio. Per fortuna potete farla scomparire passando alla modalità di visualizzazione a tutto schermo. Tornando per un attimo a TMux, non possiamo fare a meno di lodare la sua gestione dello spazio. Non c'è un solo pixel che vada sprecato in tutto l'ambiente di lavoro. Questo sia che si utilizzino solo applicazioni a riga di comando, sia che si sfrutti un browser o altri programmi in modalità multi-monitor. Il meglio di sé, comunque, TMux lo dà nell'uso delle connessioni remote. Chiunque utilizzi Rails troverà eccezionale la condivisione di una sessione TMux/Vim in remoto. Emacs, a meno che non si eseguano applicazioni da terminale, propone delle barre complete di menu con varie icone che possono sembrare ridondanti. Una volta acquisita familiarità con le scorciatoie da tastiera, si può comunque utilizzare Emacs senza l'ausilio di alcun orpello grafico.



› **StumpWM offre sempre un ambiente ben organizzato e pratico da usare per qualsiasi genere di attività**

Verdetto

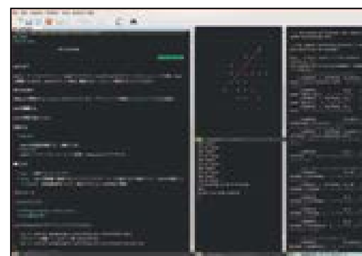
XMonad ★★★★☆
StumpWM ★★★★★
i3 ★★★★★
TMux ★★★★★
Emacs ★★★★☆

» Il meglio dei due mondi è dato da i3, che offre sia la modalità a schermo intero sia una taskbar tra le più utili.

I migliori extra

Menu, applet e molti altri strumenti

Oltre alla taskbar di i3, ci sono altre aree di notifica e stato che possono essere utilizzate con qualsiasi WM, compreso XMobar (componente originariamente scritto per XMonad). i3 e StumpWM funzionano bene con stalonetray, un'area di notifica in cui si possono collocare avvisi e icone su cui fare click per modificare il layout della tastiera, la gestione della rete, la sincronizzazione con Dropbox e molto altro ancora. **Dmenu** è un add-on essenziale per i3 e XMonad. Per l'occasione abbiamo usato anche **wmii**, provandolo con TMux che peraltro funziona anche con molti altri WM. Mentre i nostri tre Windows Manager che sfruttano le tile possono condividere tutti i menu e le barre delle applicazioni progettate per i sistemi X-Window, TMux non è in grado di farlo. Tuttavia ha molti altri assi nella manica, come svariati plug-in di monitoraggio e un incredibile numero di client a riga di comando da usare per ogni attività. **Htop**, per esempio, è progettato per funzionare con una console FGA e fornire informazioni utili sui processi in esecuzione e sull'uso delle risorse. Tuttavia si può inserire in un riquadro di TMux e lasciargli monitorare l'uso della RAM e della CPU. Emacs ha una propria visualizzazione personalizzabile.



› **Emacs vi permette di tenere sotto controllo praticamente qualsiasi cosa**

Per impostazione predefinita, nella barra di stato, sono riportate le informazioni relative ai file in fase di modifica, più quelle riguardanti le modalità d'uso del WM.

Verdetto

XMonad ★★★★☆
StumpWM ★★★☆☆
i3 ★★★★★
TMux ★★★★☆
Emacs ★★★★★

» La barra di stato presente in i3 fornisce sempre informazioni molto utili. In più è particolarmente configurabile.

Windows Manager

Il verdetto

TMux è il più utile se si utilizzano spesso sessioni SSH. Quello che però ci ha sorpreso è la gestione delle finestre. Nel momento in cui abbiamo organizzato l'ambiente di lavoro, ci siamo resi conto di quanto sia versatile, ma soprattutto non dia conto di alcuna distrazione. XMonad, invece, è forse il più semplice con cui iniziare ma manca di intuitività nei riguardi del proprio file di configurazione in Haskell. Questo, come detto, richiede una certa competenza con tale linguaggio che non tutti hanno. Nonostante ciò, usare XMonad è davvero una gioia. Si integra alla perfezione con Gnome e potrebbe essere una delle prime scelte su cui puntare. La gestione della tastiera e degli spazi di lavoro è molto semplice, tanto da essere un invito a utilizzarlo prima ancora di provare gli altri WM. StumpWM è forse il Windows Manager che sfrutta meglio la vera filosofia del minimalismo. Combina alla perfezione un'interfaccia semplice con potenti opzioni di configurazione. Come detto, per la configurazione di StumpWM è necessario avere delle basi di Lisp. A nostro parere, a differenze di XMonad, questo rappresenta un buon compromesso tra potenza e curva di apprendimento non tropo ripida. Emacs può essere amato o odiato. Vie di mezzo non ce ne sono. Se non siete dei fan, difficilmente riuscirete ad apprezzarlo a dovere. Tuttavia Emacs è altamente gestibile e nonostante l'interfaccia non ci abbia fatto particolarmente impazzire, abbiamo trovato uno degli ambienti più produttivi e potenti in circolazione, soprattutto se siete tra coloro che passano la maggior parte della giornata davanti alla riga di comando. Il vincitore, comunque, è i3 che offre un file di configurazione semplice e veloce cui metter mano. Non è forse potente come Emacs, ma è più che sufficiente per gestire quanto ci serve.



› i3 è in assoluto il vincitore di questa comparativa. È davvero difficile trovare dei punti deboli

> "i3 non è potente come Emacs, ma è facile da configurare e offre tutto ciò di cui abbiamo bisogno"

1° i3 ★★★★★

Web: i3wm.org **Licenza:** BSD **Versione:** 4.8

» Non gli manca niente ed è il più facile da configurare.

2° Emacs ★★★★★

Web: gnu.org/software/emacs **Licenza:** GPL **Versione:** 24.4

» In assoluto il più potente e flessibile. Vale la pena provarlo.

3° StumpWM ★★★★☆

Web: github.com/stumpwm **Licenza:** GPL **Versione:** 0.9.8

» Una vera gioia da usare, soprattutto senza il mouse.

4° XMonad ★★★★☆

Web: xmonad.org **Licenza:** BSD-3 **Versione:** 0.11

» Se non doveste conoscere Haskell per configurarlo, sarebbe perfetto.

5° TMux ★★★☆☆

Web: tmux.sf.net **Licenza:** BSD **Versione:** 1.9a

» La vera potenza della riga di comando. Ottimo per l'accesso remoto.

A voi la parola...

Non siete d'accordo con le nostre scelte? Avreste usato altri WM? Inviate le vostre opinioni su questo Confronto a: **recensioni@linuxpro.it**

Considerate anche...

In Rete si trovano una grande quantità di WM da provare. Un esempio oltre quelli che vi abbiamo proposto è **Awesome WM**: estremamente configurabile, si usa molto bene anche senza mouse. L'unico aspetto che forse lascia perplessi è la necessità di utilizzare il linguaggio di programmazione Lua per la configurazione. Provando **wmii**, **evilwm** e **iwm** sotto Emacs e TMux, ci siano innamorati di questi ambienti davvero minimali. Basta una veloce configurazione per ottenere uno spazio di lavoro estremamente produttivo. Infine non disdegnate le possibilità di gestione del vostro desktop. Che sia Gnome, Unity, Xfce o LXDE, ci sono sicuramente delle opzioni per configurare a dovere un ambiente produttivo e versatile. Provate per esempio a gestire i tasti di scelta rapida o a massimizzare lo spazio sullo schermo. Non ve ne pentirete. LXP

Da non perdere

» Guacamole » Dr. Geo » Zurmo » KPhotoAlbum » SWI-Prolog » Weechat » QMMP » Birdfont » Otter Browser » Rosa ImageWriter » ModemManager » Fame » 8-Bit Bantitos

Desktop remoto

Guacamole

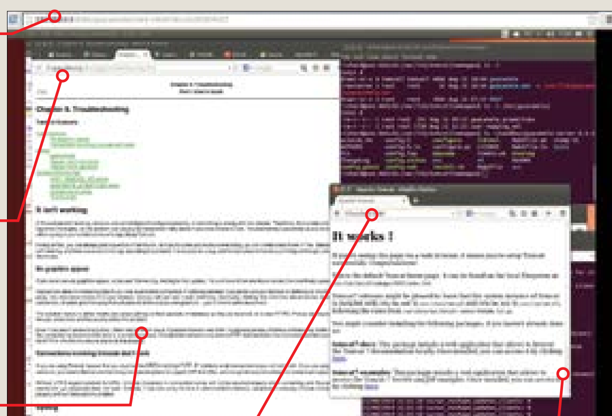
Versione: 0.9.4 **Web:** http://guac-dev.org

L'accesso remoto è una grande comodità ma non si ha sempre a disposizione un client VNC con cui lanciarlo, soprattutto se state utilizzando uno smartphone o un tablet magari preso in prestito da qualche amico. Ecco perché parliamo di **Guacamole**: un client che consente di collegarsi in remoto al vostro desktop da qualsiasi browser. Visto che molte distro distribuiscono in ritardo i vari pacchetti, abbiamo deciso di compilare Guacamole dal sorgente. Per prima cosa installate **Tomcat7** dai vostri repo, quindi recuperate i pacchetti **tar.gz** per **guacamole-server** e **guacamole-client**, oppure il file **ready-build.war**. Il pacchetto **guacamole-server** include un proxy (**guacd**) per tradurre un insieme di protocolli JavaScript e binari VNC e RDP. Probabilmente avrete già installato le poche dipendenze richieste, tranne forse la libreria **OSSP UUID**. Potrebbe essere necessario aggiornarle alle versioni più recenti. Nel caso, potete controllare il manuale di Guacamole o lanciare il **./configure** per farvi dire cosa manca. Questo comando vi informa anche su quali protocolli di sistema Guacamole utilizza. A questo punto non dovete fare altro che eseguire il classico **make** e **sudo make install**, quindi **sudo idconfig** per aggiornare le librerie.

› Non c'è alcuna GUI ma per molti aspetti non se ne sente neppure la mancanza

> **"Guacamole non è un software banale ma neppure complesso"**

L'interfaccia di Guacamole



Tutto nel browser
Guacamole permette di collegarsi da un browser senza client, traducendo VNC e RDP in un'interfaccia JavaScript.

Documentazione
Il manuale è davvero completo. Se non dovesse bastare, rivolgetevi al forum di discussione e alla comunità di SourceForge.

Configurazione
Il montaggio e la configurazione non sono banali ma non per questo particolarmente difficili. In altre parole, Guacamole è un software che ogni utente Linux dovrebbe provare almeno una volta.

Tomcat
Guacamole funziona su Tomcat ed è necessario installare i file in modo che quest'ultimo li possa trovare.

Condivisione Desktop
Se si usa un server VNC potete abilitare anche la condivisione di schermo e tastiera.

A casa e fuori

Dopo aver configurato Guacamole (per tutti i dettagli date un'occhiata al manuale) e collocato il pacchetto **.war** in **/var/lib/tomcat7/webapps** così che Tomcat possa trovarlo, siete pronti per iniziare. Se non avete voglia di compilare il tutto manualmente, niente vieta di installare una versione non aggiornata proveniente dal Software Manager. Di funzioni ce ne sono a sufficienza e crediamo siano più che appropriate per provare l'accoppiata server-client e decidere se fa al caso vostro. Una volta installato e attivato, Guacamole vi farà trovare la propria pagina all'indirizzo **http://localhost:8080/**. Sarà poi sufficiente un semplice forward per uscire all'esterno e collegarvi al vostro desktop da qualsiasi luogo.

Programma educativo

Dr. Geo

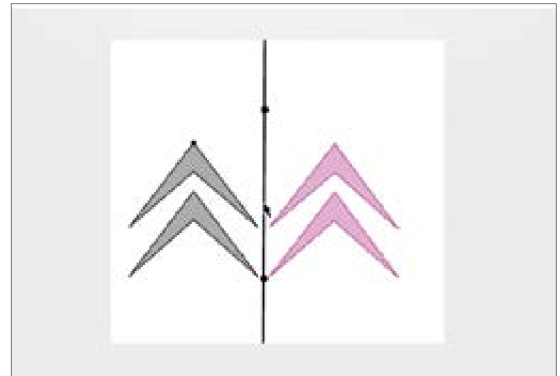
Versione: 14.12b **Web:** http://www.drgeo.eu

Dr. Geo lo troviamo spesso nelle distro educative sotto il menu **Giochi**. In realtà, seppure il suo utilizzo sia molto divertente, non si tratta di un vero e proprio passatempo.
Se utilizzato con cognizione, può essere uno strumento davvero interessante. Infatti, dopo aver preso confidenza con i principali strumenti, sarà possibile creare disegni geometrici estremamente precisi. Prima di procedere è però necessario perdere qualche minuto parlando dell'installazione. La procedura, per fortuna, è molto semplice. Esattamente quello che ci si aspetta da un prodotto oramai maturo e collaudato. Se non avete l'ultima versione disponibile nelle repo della vostra distro, sul sito ufficiale troverete un file compresso con i binari. In alternativa c'è anche il codice sorgente disponibile su Launchpad. Dr. Geo funziona bene anche su tablet e può essere scaricato sia da Google Play, sia da App Store. Quando si avvia l'applicazione, vi trovate davanti a un foglio bianco. La barra degli strumenti permetterà di trovare tutte le funzioni per iniziare a disegnare qualsiasi genere di forma, dalle più semplici alle più complesse.

I primi passi

Dopo aver lanciato l'applicazione e provato a creare qualche forma, sicuramente sentirete il bisogno di leggere un po' di documentazione. A questo proposito potrete trovare sia quella in linea, sia svariati video su YouTube che vi aiuteranno ancora meglio a capire come funziona il programma. Il software può comunque essere utilizzato dai ragazzi in età scolare. I comandi principali, quindi, non sono assolutamente complessi da imparare. Si inizia con dei punti e li si usa per definire linee, curve e forme. Con lo strumento di trasformazione si può poi creare qualcosa di molto più complesso. Noi, per esempio, ci siamo limitati a realizzare il logo Citroen che potete vedere nello screenshot qui sopra. I progetti possono essere esportati come PostScript, LaTeX o PNG, così come in formato XML.

"Dr. Geo è incluso in molte distro educative sotto il menu Giochi"



› Creare queste forme non è stato immediato, ma sono comunque bastati una ventina di minuti

CRM

Zurmo

Versione: 2.8 **Web:** http://zurmo.org

Un CRM, acronimo di *Customer Relations Management*, è un programma che tutte le piccole e medie imprese dovrebbero avere.
Si tratta di un software organizzativo che permette di indicizzare i clienti e tenere traccia di tutti i contatti che si hanno con loro. **Zurmo** è un CRM e ha tutte le caratteristiche che ci si aspetta da un'applicazione di questo genere: gestione del prodotto, report personalizzati, avvisi, anagrafica e molto altro ancora.
L'installazione è piuttosto semplice.
Per prima cosa dovrete procurarvi Apache, preferibilmente con **mod_deflate**, quindi Memcached e MySQL Server 5.1 o superiore. Non deve poi mancare il supporto per PHP con circa una dozzina di estensioni tra cui **ctype** e **mbstring**.
Una volta installato e funzionante, Zurmo vi lascerà stupiti per l'intuitività con cui riuscirete a utilizzarlo.

La ludicizzazione

Il punto di forza di Zurmo, però, è un altro. Questo software è stato progettato utilizzando la così detta *Gamification*. Con questo termine si indicano i programmi che, pur non essendo giochi, sono stati progettati utilizzando elementi provenienti dal mondo dei videogame. Gli sviluppatori di Zurmo hanno ben pensato di mutuare alcuni elementi provenienti dall'universo videoludico, inserendoli nel proprio programma per renderlo così ancora più pratico da utilizzare. In Zurmo, per esempio, in base alle attività di un cliente, questo guadagna un determinato punteggio, gli vengono assegnati dei distintivi che lo contraddistinguono come un buon o cattivo committente. Ci sono poi le classifiche e molti altri elementi che troviamo in qualsiasi videogioco. Purtroppo non abbiamo avuto il tempo materiale di provarlo a fondo. Per farlo sarebbero necessari svariati giorni e una banca dati di clienti da inserire nel database. Tuttavia quello che abbiamo letto ci è sembrato davvero interessante e pertanto, se avete una piccola attività, vi consigliamo di prenderlo in seria considerazione.

"Zurmo sfrutta un sistema di ludicizzazione per gestire i clienti"



› Zurmo permette di tenere traccia di tutte le attività che abbiamo compiuto per un determinato cliente

Organizzatore di immagini

KPhotoAlbum

Versione: 4.5 **Web:** http://kphotoalbum.org

Nei vari "Da non perdere" che si sono succeduti nel tempo abbiamo sempre trattato diversi gestori d'immagini. Ci siamo soprattutto concentrati su quei programmi che supportano l'importazione, la manipolazione e la modifica delle foto. Per alcune persone, però, questo genere di applicazioni è troppo dispersivo. Soprattutto se il nostro intento è organizzare una semplice collezione di immagini. Aprite il vostro Software Center e scrivete **KPhotoAlbum**. I risultati non tarderanno ad arrivare. I pacchetti dovrebbero essere già aggiornati, ma se così non fosse, sul sito ufficiale del programma è disponibile tutto il necessario per compilare il sorgente. Non appena si avvia KPhotoAlbum per la prima volta, è possibile scegliere se inizializzare il proprio database da una precedente installazione, crearne uno nuovo o eseguire la demo che permette di provare la codifica e le funzionalità di ordinamento. Il programma ci consiglia poi di leggere il manuale d'uso che potrebbe rivelarsi interessante, soprattutto per scoprire alcuni comandi avanzati. Essendo un software KDE, tutta la documentazione è presente sotto forma di manuale. Dal menu **Guida** è possibile eseguire una demo che prevede il collegamento di un dispositivo Android, da cui sarà possibile visualizzare le immagini presenti nel programma (si tratta di una delle principali novità di questa versione). Indipendentemente dal fatto che vogliate o meno leggere la documentazione, i principali comandi di KPhotoAlbum sono davvero semplici da usare. Una volta importate le foto, sarà possibile raggrupparle con dei tag, inserirle in album e suddividerle per categoria. Come Digikam, KPhotoAlbum beneficia di un'architettura modulare KDE: possiamo sfruttare numerosi plug-in per l'importazione, l'esportazione e la manipolazioni di vaste raccolte fotografiche. Inoltre potremo ordinare e catalogare rapidamente gli album.

> **"KPhotoAlbum è perfetto se avete bisogno di catalogare le vostre foto"**



› **KPhotoAlbum dispone di un'eccellente funzione di tagging, in più la versione 4.5.0 introduce la marcatura di alcune aree**

Ambiente per la programmazione logica

SWI-Prolog

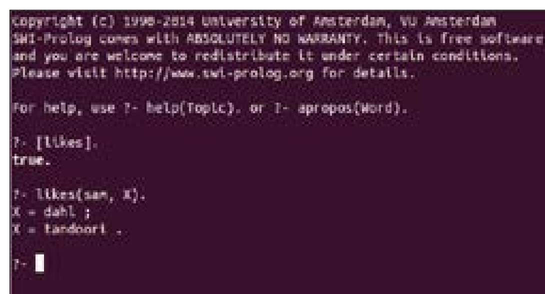
Versione: 7.1.28 **Web:** www.swi-prolog.org

Prolog è un linguaggio di programmazione logico dichiarativo. Al giorno d'oggi non è molto utilizzato, anche se può vantare una certa influenza nel mondo del design di Erlang. Se volete approfondire la conoscenza con Erlag e la sua strana sintassi, allora **SWI-Prolog** è un ottimo punto da cui partire. Sviluppato da Jan Wielemaker presso l'Università di Amsterdam nel 1987, viene fornito con un terminale di comandi e varie altre interfacce. Eseguendo:

```
% sw
?- emacs. .
```

si apre **PceEmacs** (un clone di Emacs) in una finestra separata, dandovi il controllo della sintassi. Potrete poi utilizzare anche l'interfaccia multipiattaforma GUI toolkit XPCE che funziona anche con Lips e C++. C'è poi JPL, un'interfaccia bidirezionale tra Java e Plorog. Il pacchetto include anche un plug-in Eclipse, **PDT**. Oltre a fornire pacchetti per le distribuzioni basate su Debian e Ubuntu (compreso un PPA per Ubuntu) ci sono quelli per Red Hat (Fedora, RHEL, CentOS), SUSE e Mageia. Nel caso, in Rete troverete tutta la documentazione necessaria per creare pacchetti adatti alla vostra distribuzione. A questo proposito è opportuno aprire una parentesi sul supporto che la comunità offre. Dando un'occhiata al sito, non si può fare a meno di notare una grande quantità di risorse. La documentazione è suddivisa in vari comparti, tra cui il manuale e le FAQ. Ci sono poi una serie di tutorial organizzati per principianti, utenti intermedi e avanzati. Infine troviamo perfino una wiki fornita più che discretamente. In definitiva, avete a disposizione tutti gli strumenti per imparare a usare SWI-Prolog con la massima soddisfazione e una buona curva di apprendimento. Considerando le tante funzioni e le potenzialità di questo ambiente di programmazione, possiamo dire che si rivolge soprattutto a chi cerca di costruire programmi reali e non solo a chi progetta esercitazioni accademiche.

> **"SWI-Prolog si rivolge anche a chi lavora con l'intelligenza artificiale"**

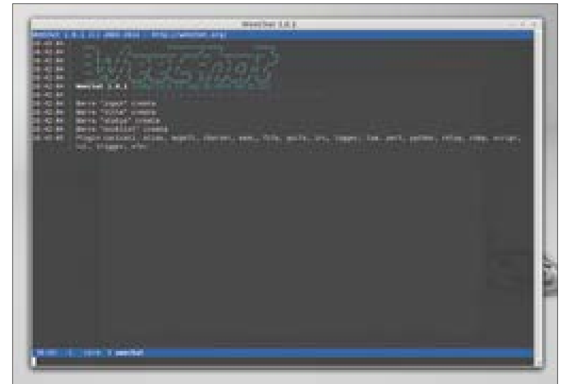


› **Ecco qui una prima query eseguita con SWI-Prolog. Un passo avanti rispetto al tradizionale "Ciao Mondo!"**

Client Chat

Weechat

Versione: 1.0.1 **Web:** http://weechat.org



› Dopo aver installato tutte le dipendenze, per avviare Weechat è sufficiente aprire il terminale e inserire il comando weechat

Nonostante gran parte delle relazioni sociali che corrono sul Web si sia spostato su Twitter, Facebook e Google Plus, molti sviluppatori continuano a lavorare sulla creazione di chat nei canali IRC. Per la gestione e l'accesso a questi luoghi di discussione ci sono programmi di ogni genere. Purtroppo, quelli integrati nella maggior parte delle distro non brillano per leggerezza e funzionalità. Ecco perché abbiamo puntato l'attenzione su Weechat. Semplice e intuitivo, questo programma ha tutto quello che ci si aspetta da un software a riga di comando basato su ncurses.

Tweet e Chat

Al primo avvio è possibile specificare il server e il canale a cui collegarsi. Per farlo è sufficiente inserire i seguenti comandi, sostituendo naturalmente il nome del server e del canale con quelli appropriati

```
weechat -r irc://tuonickname@irc.freenode.net/#nomecanale
```

Il manuale è disponibile per una consultazione in forma completa o rapida. In quest'ultimo caso vi fornisce le basi e i comandi per iniziare a usare il programma. Probabilmente sarà difficile ricordare tutte le istruzioni fin dal principio. Vi consigliamo pertanto di tenere aperta una seconda finestra con il manuale a portata di mano. In questo modo potrete inserire i vari comandi senza troppi pensieri. La velocità e la flessibilità di Weechat vi faranno dimenticare che non state utilizzando una chat dotata di interfaccia grafica. Detto questo, sono comunque disponibili diverse GUI, tra cui quelle in HTML5, Javascript e client Android. Con IPv6, Socks5, SASL, certificati SSL e SSH, Weechat può funzionare come un proxy IRC. Tra le sue funzioni troviamo il controllo ortografico e una piena estensibilità tramite script Python, Lua, TCL, Guile, Ruby, Perl e C. In realtà, se vogliamo sbizzarrirci, sul sito ci sono ben 280 script pronti all'uso.

> **"La flessibilità di Weechat vi farà dimenticare di non avere una GUI"**

Audio Player Qt

QMMP

Versione: 0.8.3 **Web:** http://qmmp.ylsoftware.com



› Non distraetevi troppo con le opzioni per la gestione della skin. Concentratevi invece sull'audio

I player Audio **QMMP** è un programma multipiattaforma basato su Qt. Il suo progetto deve molto a WinAmp e XMS, da cui riprende l'interfaccia utente. A parte questo, comunque, gli sviluppatori hanno messo in campo un progetto completamente nuovo, mirato a funzionare bene. L'installazione è semplice, soprattutto se si sta utilizzando una delle tante distribuzioni che mettono a disposizione i pacchetti. Come sempre, nel caso non fossero aggiornati, colleghiamoci al sito del programma. Al primo avvio veniamo accolti da un'interfaccia semplice e intuitiva. Tramite i menu e le scorciatoie da tastiera possiamo poi gestire tutte le funzioni principali. Ci vuole poco a sentirsi a proprio agio e basta ancora meno per raggruppare i file audio in playlist organiche. Parte del successo di QMMP sta nelle funzionalità disponibili nelle finestre separate dalla GUI. In questo modo possiamo chiuderle quando desideriamo, risparmiando spazio sul desktop. Naturalmente è possibile aprire l'equalizzatore grafico, il visualizzatore audio e molte altre opzioni tipiche di un player audio. Se volete un po' di ispirazione, potete dare un'occhiata ai vari screenshot presenti sul sito ufficiale di QMMP. Dopo aver personalizzato la grafica, è il momento di passare all'installazione del QMMP Pluglin Pack, che contiene FFap e il decoder Monkey's Audio (APE) con campioni a 24-bit e il supporto per CUE, MPEG v1/2/ layer1/2/3 decoder.

> **"QMMP è un player audio estremamente versatile e potente"**

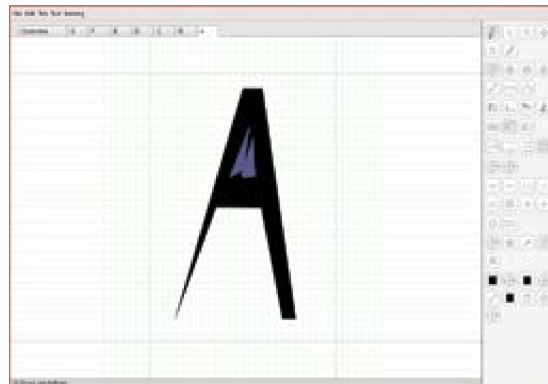
Flessibilità

In aggiunta al pacchetto per il supporto predefinito, troviamo la possibilità di gestire Ogg Vorbis/Opus, FLAC, Musepack, WavePack, MOD files, AAC, Audio CDs, WMA, PCM, MIDI, SID e file Chiptune. Se non dovesse bastare, ci sono anche i plug-in **DSB** (effetto bs2b, conversione della frequenza di campionamento, effetti **LADSPA** e molto altro), il download dei testi, visualizzatori ed equalizzatori di ogni genere, interfacce per le API audio più comuni e via dicendo.

Editor di font

BirdFont

Versione: 1.9.3 **Web:** http://birdfont.org



› L'interfaccia di BirdFont è un vero piacere. Permette di realizzare qualsiasi progetto si abbia in mente

Birdfont è un editor di font con licenza GNU-GPL. Permette la creazione grafica e vettoriale di immagini che possono essere esportate in formato TTF (*True Type Font*), EOT o SVG. Inoltre incoraggia all'uso della licenza SIL Open Font. La versione 0.1 è stata rilasciata appena due anni fa e visto il numero di release a cui ci troviamo adesso, non possiamo fare a meno di lodare l'attività degli sviluppatori. Le dipendenze necessarie sono GTK/pango/Cairo da Gnome. Dando un'occhiata ai vostri Software Manager, non dovreste avere difficoltà a trovare l'ultima versione di Birdfont. Questa applicazione è scritta in Vala, un linguaggio orientato agli oggetti di Gnome, che offre funzionalità di tipo C# per runtime C con un overhead minimo. Si tratta di un progetto molto vasto che coinvolge parecchi sviluppatori e ne cerca altrettanti. Se avete voglia e competenze, potreste chiedere di farne parte. Peraltro sono alla ricerca di persone che possano scrivere tutorial, disegnare icone, gestire il sito Web, le traduzioni e molto altro ancora.

A, B, C...

I tutorial per i principianti sono disponibili sul sito Web e su **http://mistifonts.com**. C'è parecchio fermento e scommettiamo che questo software continuerà a migliorare nel tempo così come ha fatto fino a ora. Durante l'uso del programma non aspettiamoci dei risultati immediati. La creazione di un font, in generale, è un'operazione certosina che richiede una gran dose di pazienza e umiltà. Molto spesso penserete di essere sulla strada giusta, invece dovrete cancellare e ricominciare da capo. In altre parole, Birdfont non è uno strumento magico che vi permetterà di creare il vostro font in cinque minuti. Dovrete imparare a conoscerlo e a utilizzarlo a dovere. Solo così potrete ottenere risultati degni di nota. Tuttavia è molto meno complesso di altri software simili.

> **"Creare un font è un'operazione che richiede molta pazienza"**

Web browser

Otter Browser

Versione: 0.9.04 **Web:** http://otter-browser.org



› Il browser, messo alla prova con HTML5, sfoggia dei risultati decisamente buoni. Niente male per una beta

Otter Browser è un progetto in Qt5 impegnativo e piuttosto coraggioso. L'obiettivo è ricreare un'esperienza simile a quella offerta da Opera 12.x. Come sapete, nel 2012, Opera Software ha interrotto il supporto per la propria tecnologia di base, vale a dire il motore di rendering Presto HTML, il cuore del desktop e del browser mobile Opera. Il team di Opera Software si è largamente ridotto e così l'azienda ha deciso di spostare i suoi sforzi sul progetto Chromium, aiutando Google a sviluppare il codice per il suo browser Open Source. Da allora il "nuovo" Opera è basato sul motore Blink di Google che naturalmente lo fa assomigliare molto a Chromium. Infatti, le caratteristiche più amate di Opera Presto sono state scartate per realizzare un clone ridisegnato di Chromium. Le funzioni di cui stiamo parlando sono tante e comprendono la gestione dei preferiti, quella dei contenuti integrati, lo switcher Javascript e molto altro ancora. Gli sviluppatori di Otter Browser, quindi, sono intenti a ricreare tutte quelle funzioni di spicco che hanno fatto di Opera uno dei migliori browser in circolazione. Il progetto è ancora in fase di beta e tra i suoi successi vanta un supporto iniziale per l'estensione Adblock Plus, per l'importazione dei segnalibri HTML, la personalizzazione della tastiera e dei tasti di scelta rapida, un download manager, un gestore di cookie, una pratica sidebar e molti altri miglioramenti di minor entità all'interfaccia utente. Otter Browser è l'unico a essere basato interamente sul widget Qt5. Il programma è disponibile per una grande quantità di distribuzioni, tra cui OpenSUSE, Ubuntu, Arch e i loro rispettivi derivati. Il progetto si sta comunque sviluppando molto rapidamente. I programmatori pubblicano rapporti settimanali sul loro sito Web. Se siete interessati, potete installare la beta e seguire con costanza gli sviluppi sulle pagine di Otter Browser.

> **"Otter Browser è pensato per riproporre le funzioni di Opera 12"**

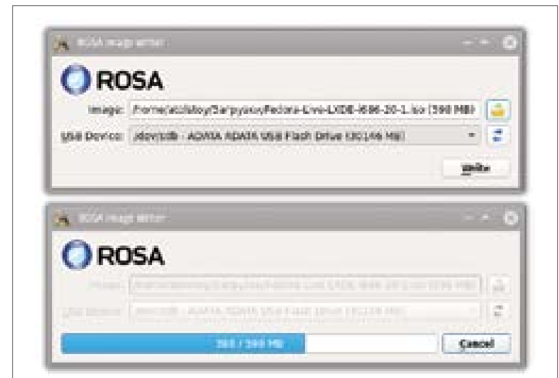
Strumento per la creazione di supporti USB live

Rosa ImageWriter

Versione: 2.5.0.0 **Web:** http://bit.ly/ZgurWL

I dischi ottici stanno rapidamente lasciando il nostro mondo. La completa sostituzione con le unità flash USB e il Cloud storage è oramai solo questione di tempo. Anche se possiamo spostare i nostri file su uno spazio online, DVD e CD continuano a essere indispensabili per l'installazione dei sistemi operativi (almeno per ora). Anche in questo settore, però, i supporti USB si stanno facendo sempre più spazio. Infatti, niente vieta di usare una chiave per memorizzare la ISO di una distro Linux e procedere poi all'installazione. L'operazione può essere molto semplice se siamo abbastanza esperti, ma se non abbiamo mai fatto niente del genere, può nascondere alcune insidie. Anche se molti di voi avranno senza dubbio sentito parlare di UNetbootin, non tutti sono riusciti a utilizzarlo con successo, soprattutto al primo colpo. Infatti, è bene sapere che non tutte le ISO possono essere scritte in una memoria USB. Infatti, alcune sono progettate per CD/DVD, altre hanno bisogno di modifiche supplementari per il supporto UEFI, altre, invece, hanno un partizionamento non standard. **Rosa ImageWriter** è un componente di Rosa Linux che a sua volta è un'evoluzione di Mandriva SA Linux. Alcuni anni fa, il team di Rosa creò una fork chiamata SUSE Studio Imagewriter, uno strumento modesto e molto facile da usare per scrivere immagini ISO su dispositivi USB. La fork è poi stata fortemente migliorata: si è eliminata tutta la parte in C e .NET dalle versioni per Linux e Windows e il tutto è stato sostituito con C++. L'interfaccia è stata riscritta utilizzando Qt5, facendo però in modo di mantenere un alto grado di semplicità. Il risultato finale è uno strumento che aggiorna costantemente l'elenco dei supporti USB disponibili, in più supporta svariate versioni di ISO. Rosa ImageWriter è distribuito come tarball contenente i sorgenti, più una serie di pacchetti binari per Linux 32 e 64-bit, Windows 7/8 e Mac OS X.

> **"Rosa ImageWriter si aggiorna in continuazione con nuovi supporti USB"**



› Un'altra piccola utility Qt5 creata artigianalmente, capace di funzionare davvero egregiamente

Connection Manager

ModemManager

Versione: 1.4 **Web:** http://bit.ly/1pLSpzy

Nonostante la svolta dovuta all'uso dei dispositivi mobile, forse non ci rendiamo conto che nella maggior parte delle occasioni continuiamo a usare i tradizionali modem. In ogni smartphone è contenuto un modem, così come nei dongle USB. I giorni in cui su Linux si giocava con il comando ***ifup*** e ***ppp*** sono finiti e **ModemManager** riflette il cambiamento. Si tratta di uno strumento per la gestione di modem, dongle, smartphone Bluetooth o attrezzature ancora più professionali che necessitano di alimentazione dedicata. ModemManager è un demone *dbus-activated* che controlla la banda larga tramite dispositivi mobili e connessioni (2G/3G/4G). In più fornisce alcune funzionalità per la gestione delle chiamate e degli SMS, così come dei contatti. Se si desidera avere una connessione Internet su computer utilizzando uno smartphone come modem (ammesso che abbiate un adeguato piano tariffario) e la connessione Bluetooth o USB per il collegamento, allora ModemManager è quello che fa per voi. Il programma è fornito di una GUI che può essere lanciata come qualsiasi applicazione standalone. Inoltre viene richiamata automaticamente quando si associa il telefono tramite Bluetooth e si sceglie di aggiungere una DUN (*Dial-up networking*). In questo modo il collegamento viene aggiunto alla configurazione NetworkManager e visualizzato tra quelli già disponibili. La maggior parte degli sviluppatori Linux considera ModemManger un must da avere. Questo significa che la versione più aggiornata dovrebbe già essere disponibile nel Software Manager della vostra distribuzione. La release 1.4 può gestire lo stato di alimentazione del dispositivo, ha un miglior supporto degli IP (per esempio, adesso è possibile impostare un indirizzo IPv6 statico con modalità DHCP) e supporta i modem più recenti come ZTE, Huawei e Sequans communication.

> **"ModemManager è considerato un vero must dai professionisti"**



› La GUI di ModemManager è ben fatta e permette di configurare nel dettaglio tutti i parametri

Giochi Applicazioni per il tempo libero

Gioco di ruolo Fantasy

Fame

Versione: 0.9.6 **Web:** http://bit.ly/FameRPG

Il gioco fantasy precedentemente noto come **Untitled** è stato sotto sviluppo per oltre un decennio. Tutt'oggi, però, lo sviluppatore dichiara di non essere ancora soddisfatto e quindi non considera conclusa la fase di creazione. Ciononostante, è completamente giocabile, anche se va considerata la presenza di numerosi bug. C'è pure un buon manuale utente. Il creatore, Piotr Bednaruk, afferma che giocare non è l'unico divertimento. Il programma può essere espanso creando nuovi percorsi attraverso l'editor integrato. Inoltre è possibile generare nuove missioni con un linguaggio di scripting dedicato. Il binary funziona praticamente su qualsiasi macchina. Basterà scompattare il file ed eseguire **./Fame**.

Dopo una breve personalizzazione del personaggio (i valori predefiniti vanno bene per iniziare), è possibile lanciarsi nella prima missione: trovare Asrigam in paese e accettare la sua offerta per la caccia al mostro. Michael Toy e Glenn Wichman, nel lontano 1980, progettarono Rogue che ha ispirato molti cloni e giochi simili. **Fame** è uno di questi.

Si installa al volo

Uno dei punti di forza di Fame è lo scripting UG estremamente documentato, il quale consente di personalizzare ed estendere il gioco in molteplici modi. Per esempio, è possibile scrivere uno script per gestire il dialogo con i personaggi non giocanti. Inoltre potrete divertirvi a modificare il comportamento dei mostri, alterare i commerci e aggiungere ulteriori script di localizzazione. Ogni aggiunta, poi, può essere resa permanente o limitata nel tempo.

> **"Fame si rifà con orgoglio alla tradizione iniziata con Rogue"**



› **Sferrare un attacco a un mostro è uno dei primi compiti che dovrete affrontare in questo gioco**

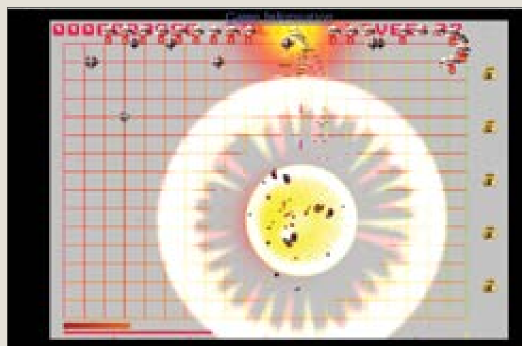
Gioco arcade su browser

8-Bit Bantitos

Versione: 1.2 **Web:** http://bit.ly/8-BitBantitos

Steven De Toni, il creatore di questo gioco, ha deciso di riprendere in mano le redini dello sviluppo dopo aver lanciato il titolo circa un anno fa. **8-Bit Bantitos** è liberamente ispirato ai classici del Commodore 64 ed è perfettamente all'altezza di questi grandi classici del mondo videoludico. È davvero divertente, sia che si voglia intraprendere il single player, sia l'avventura online su **http://a550511.magix.net**. Il gioco sembra davvero retrò, ma dando un'occhiata alla programmazione si notano diverse implementazioni aggiornate. Infatti, per funzionare a dovere, 8-Bit Bantitos ha bisogno di una versione piuttosto recente di Chrome. Il codice è in JavaScript semplice (De Toni ha volutamente puntato sull'intuitività della codifica, così da lasciare la possibilità ad altri di mettere mano al sorgente e personalizzarlo secondo necessità). Non molto tempo fa, per scrivere un gioco del genere, avremmo avuto bisogno di C o C++. Adesso, invece, è tutto molto più semplice sia per la scrittura sia per l'eventuale modifica futura da parte dello sviluppatore o di altre persone. Gli effetti sonori sono buoni, così come la grafica e l'azione in generale. I controlli possono essere gestiti tramite la tastiera: **QAX** per il fuoco e **WSX** per lo scudo, mentre i tasti freccia per il movimento. Il segreto, però, è sopravvivere utilizzando il mouse. Il gioco può funzionare anche con un joypad o un joystick. Le FAQ sottolineano poi la necessità di imparare a utilizzare lo scudo come se fosse un arma. Nel complesso è un passatempo arcade piuttosto divertente, da giocare durante le pause. LXP

> **"Giocate 30 livelli di azione in puro stile arcade mozzafiato"**



› **Boom! Godetevi alla grande 8-Bit Bantitos su un browser con HTML5 e JavaScript**

Tutorial

I nostri esperti offrono ogni mese i loro consigli di programmazione e di amministrazione del sistema

LA VOSTRA GUIDA DI RIFERIMENTO

Esiste sempre qualcosa di nuovo da imparare in campo informatico, soprattutto in un mondo dinamico come quello di Linux e dell'Open Source. Ogni numero di Linux Pro presenta una corposa sezione dedicata a tutorial realizzati da esperti in moltissimi settori: programmazione, sicurezza, amministrazione di sistema, networking. Troverete informazioni utili sia che siate dei veterani di Linux sia degli utenti alle prime armi. Studieremo con cura anche le applicazioni più diffuse sia in ambito lavorativo che desktop. Il nostro scopo è quello di fornire in ogni numero il giusto mix di argomenti, ma se avete suggerimenti su temi particolari che vorreste vedere trattati, scriveteci via e-mail all'indirizzo tutorial@linuxpro.it

COME RAPPRESENTIAMO LE LINEE DI CODICE

Si presenta spesso la necessità di riportare le linee di codice di un programma. Per favorirne la lettura evidenzieremo le singole linee in questo modo:

```
begin
  mniWordWrap.Checked := not
end
```

Quando una riga di codice supera la lunghezza della colonna la riporteremo su più righe utilizzando la notazione seguente:

```
printf("Vi preghiamo di inserire
una password.");
```

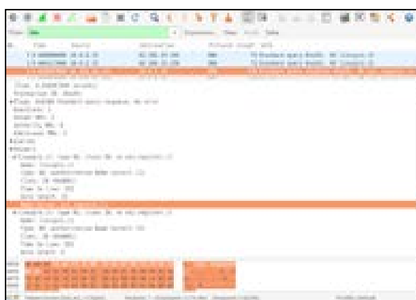
TUTORIAL



Fondamenti di HTML

- Home
- Ricerca

Benvenuti nel tutorial sull'HTML e CSS3 di Linux Pro





ACCADEMIA DEL CODICE

Jenkins e Dockerfile

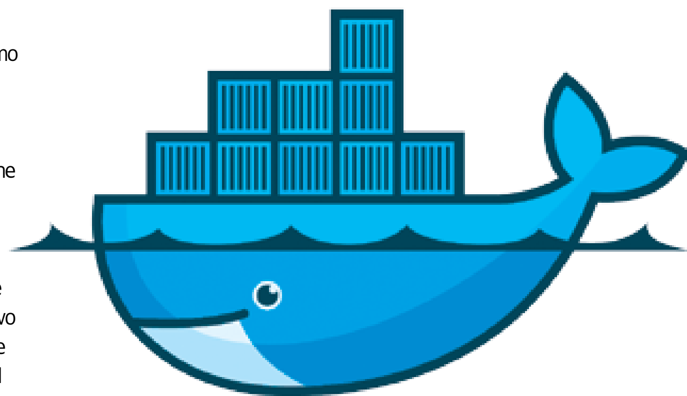
Lo staff di Linux Pro dà un'occhiata ad alcune attività collegate all'adozione di Docker in un ambiente di sviluppo, inclusa l'integrazione continua con Jenkins



Nel tutorial del mese scorso abbiamo parlato di Docker, un software per implementare contenitori in Linux, e abbiamo visto alcune funzioni di base e semplici comandi. Partendo da ciò, andremo a fare qualche passo avanti verso l'adozione di Docker in un ambiente di sviluppo. Vedremo delle opzioni per condividere i contenitori Docker con i vostri colleghi e vedremo anche come Docker può essere usato per un workflow a integrazione continua (CI) utilizzando il famigerato tool **Jenkins**. Prima daremo una rapida occhiata ad alcune attività che un amministratore di sistema preferisce fare prima di eseguire qualunque servizio: cose come fare backup di tutto, gestire i log, ecc... Non parleremo di nuovo dell'installazione di Docker, se non siete sicuri di come eseguirla date un'occhiata al numero precedente oppure alle semplici istruzioni del sito **www.docker.com**. Le aziende che offrono servizi IT di solito hanno diversi ambienti nei quali eseguire la stessa applicazione nelle diverse fasi dello sviluppo. L'ambiente 'Dev' potrebbe corrispondere al laptop personale di ogni sviluppatore per esempio. L'ambiente di produzione o 'Live' è l'ambiente finale. Fasi intermedie potrebbero essere UAT (*user acceptance testing*, test di accettazione utente), DR (*disaster recovery*) o Pre-Prod (molto simile alla produzione, utilizzata a volte per testare dei fix al programma). All'interno di questi ambiente possono risiedere diverse versioni dell'applicazione (si spera siano in ordine lineare, ma spesso non è così). Nelle infrastrutture datate ognuno di questi ambienti consiste in una scatola fisica con una sua installazione completa di Linux e un suo disco locale. Mantenere tutti questi server può costituire un vero mal di testa per gli amministratori di sistema. Gli ambienti dovrebbero idealmente essere identici per essere sicuri che l'applicazione ospitata si comporti sempre allo stesso modo. Nonostante tutti i tool a disposizione degli amministratori di sistema, la familiare frase "ma sulla mia macchina funzionava" echeggia ancora in giro per il mondo. Docker è la soluzione di questo problema permettendo all'app di essere assemblata da componenti all'interno di un contenitore eseguibile in tutti gli ambienti desiderati.

Utilizzare i Dockerfiles

Nel primo tutorial abbiamo utilizzato molti comandi da riga di comando per gestire i contenitori di Docker, nella realtà praticamente tutti gli sviluppatori che utilizzano Docker utilizzano i Dockerfiles. Questi



semplici file di testo offrono il beneficio di essere facilmente inseribili in un sistema di controllo di versioni (possiamo salvarli in **Git** o qualunque altro sistema di controllo dei sorgenti desideriamo) e oltre a essere di costruzione più semplice rispetto agli script da shell sono molto potenti. Ecco un esempio di un file che produce una pila Ruby on Rails completa:

```
FROM phusion/passenger-ruby21:0.9.12
ENV HOME /root
CMD ["/sbin/my_init"]
RUN gem install rails
RUN cd $HOME; rails new lxp
RUN apt-get install git -y
RUN apt-get clean && rm -rf /var/lib/apt/lists/* /tmp/* /var/ tmp/*
RUN gem install railties
RUN gem install bundler
RUN cd $HOME/lxp; bundle install
```

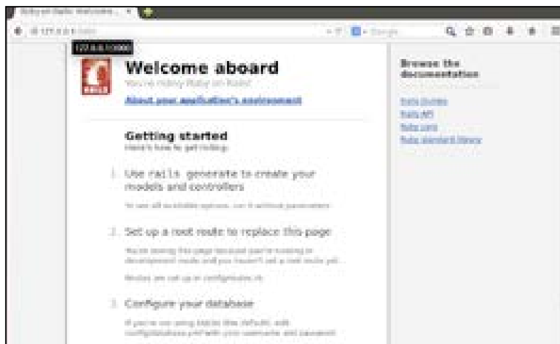
Per provare il Dockerfile, createlo in una nuova directory (vuota), il nome del file deve necessariamente essere Dockerfile. E lanciate in quella directory questo comando:

```
sudo docker build -t railstest .
```

Questo comando scarica l'immagine base dall'hub Docker, il repository di immagini di Docker Inc, e applica le variazioni necessarie a ottenere la versione specificata (nel nostro caso la 0.9.12). **Passenger-ruby** è un'immagine Docker creata dall'arcinoto (negli ambienti Ruby almeno) team di sviluppo Phusion (Passenger è un'applicazione server Web conosciuta principalmente come host per le app Ruby on Rails ma fa molto di più). La loro immagine offre molte modifiche a quella di default. Abbiamo aggiunto i comandi **gem install rails** e **cd $HOME; rails new lxp**. Chiunque abbia installato Rails di recente sa che è un lavoro abbastanza oneroso con svariati comandi da lanciare. Docker gestisce tutto questo con semplicità, grazie alla possibilità di riutilizzare l'immagine passenger (anche se questo può significare un po' di attesa per il primo download). Dopo che il download e l'installazione sono completati, possiamo lanciare un contenitore Docker con il comando seguente:

```
sudo docker run -p 3000:3000 --name lxp -t -i railstest /bin/bash
```

Questo comando avvia Docker, aggancia la porta 3000 del contenitore alla stessa porta della macchina locale, chiama



› **Rails, in Docker. Potrà non sembrare molto, ma la schermata iniziale di Twitter era simile a questa un po' di tempo**

Tip

Potete trovare il manuale completo dei comandi supportati da Dockerfile insieme a molti esempi su **http://docs.docker.com/reference/builder** (in inglese).

il contenitore **lxp**, apre un **tty**, rende il contenitore interattivo (altrimenti si chiuderebbe una volta conclusa l'esecuzione), specifica di utilizzare la nostra immagine di test di rails e per finire ci catapulta in un prompt **bash** all'interno del contenitore stesso. Da qui possiamo lanciare **rail**. Nel Dockerfile abbiamo chiesto a rail di creare un'app dentro a **/root/lxp**. Se entriamo in quella directory con il comando **cd** possiamo lanciare il comando **rails server**:

```
cd /root/lxp
rails server
```

Rail è configurato di default per utilizzare **WEBrick**, un piccolo server HTTP ruby molto leggero ottimo per gli ambienti di sviluppo. Viene avviato sulla porta 3000. Dato che abbiamo collegato quella porta del container con la stessa nel nostro host, possiamo connetterci a ruby dal nostro desktop usando l'indirizzo **http://127.0.0.1:3000**. Comparirà la familiare schermata di default di Rails (come potete vedere nell'immagine della pagina precedente).

La forza di Docker

Anche se non vi avrà impressionati molto, la forza di Docker viene dalla capacità di prendere il contenitore che abbiamo creato e utilizzarlo in posti diversi. Anche se abbiamo fatto il minimo indispensabile in termini di configurazione di Rails, potremmo aggiungerci tutto quello che vogliamo per creare un ambiente per il nostro sviluppo. Esistono un po' di opzioni per condividere le immagini create in questo modo. Docker Inc offre il suo Hub – **https://hub.docker.com** (potete vedere le immagini presenti all'indirizzo **http://registry.hub.docker.com**) che offre un'opzione gratuita (esistono opzioni a pagamento per i repository privati multipli). Comunque, in alcune situazioni il codice non deve uscire dall'azienda. In questo particolare scenario, le immagini possono essere salvate come normale file che può essere copiato direttamente sulla macchina che lo deve utilizzare e lanciato in locale. Vediamo come salvare una copia dell'immagine di test di Rails in una directory locale:

```
sudo docker save –o ./railstest railstest
```

Il file di output (sostanzialmente un archivio TAR) può essere preso e spostato ovunque vogliate. Se volete eseguirlo in una macchina diversa con Docker installato basta utilizzare il comando **load**.

```
sudo docker load –i railstest
```

Da qui possiamo fare gli stessi identici passi di prima per lanciare il server Rail nel nostro nuovo sistema. Durante il test abbiamo passato questa immagine da Ubuntu 14.04 desktop a Centos 6.5 senza alcun problema. Esiste anche un'altra opzione, eseguire registry di Docker all'interno di un data center o di una rete locale. Come potreste aspettarvi, è tutto Open Source, disponibile gratuitamente al download e facile da usare quanto un'immagine Docker.

```
sudo docker pull registry
```

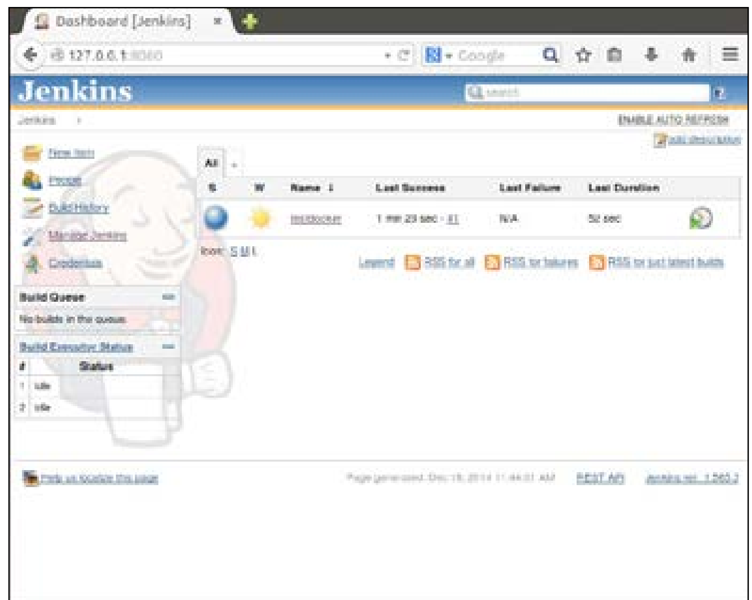
Registry ha molte opzioni: può utilizzare un database come il modulo Swift OpenStack per salvare immagini Docker, ma di default utilizza i file locali. Vediamo ora per il nostro test di dargli un percorso locale. Iniziamo con:

```
sudo docker run –e GUNICORN_OPTS=[--preload] –p 5000:5000 –v /tmp/registry:/tmp/registry registry
```

Questo comando avvia il nostro Registry locale e lo mette in ascolto sulla porta 5000. Dice anche a Docker di utilizzare una directory locale come volume collegato, i dati scritti in quella directory rimarranno persistenti anche quando il container verrà chiuso.
Ora possiamo salvare le immagini in quel container, a partire dalla nostra immagine di sviluppo railstest. Per salvare un'immagine usiamo il comando **docker push**. Di default questo comando invia l'immagine al repository globale di Docker. Per utilizzare il nostro locale, dobbiamo 'taggare' l'immagine con hostname/ip e porta.
In un nuovo terminale (dato che Docker Registry è in esecuzione nella finestra aperta), lanciamo quanto segue:

```
sudo docker tag railstest localhost:5000/railstest
sudo docker push localhost:5000/railstest
```

Questi comandi genereranno un sacco di attività in entrambe le finestre, uno stream di comandi HTTP PUT in quella con Registry in esecuzione e barre di stato di upload nell'altra. Una volta completato, eseguendo il comando **sudo docker images** dovrebbe mostrare la nostra nuova immagine **localhost:5000/railstest**. Possiamo vedere anche che Registry ha utilizzato il suo repository locale



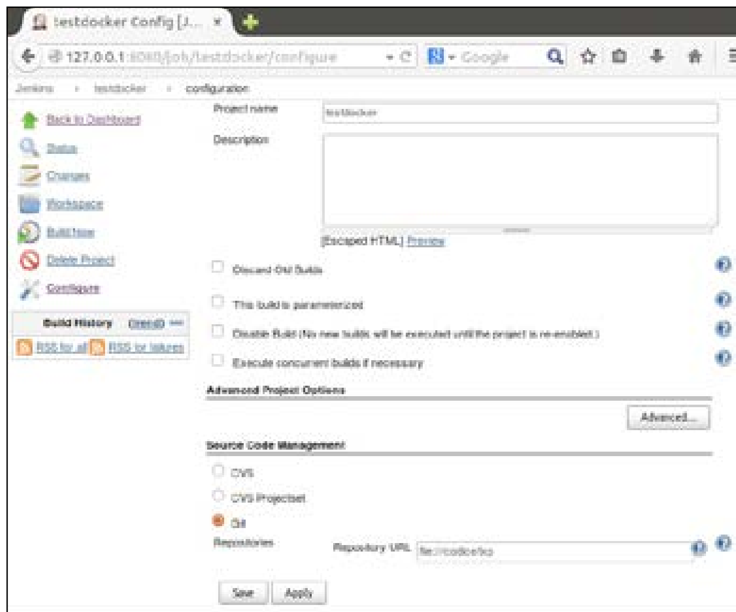
❯ **Jenkins, un tool CI con moltissime opzioni e plug-in (qualcuno potrebbe dire che sono persino troppi effettivamente)**

Tip

Se non avete dimestichezza con **vi** invece di modificare i file da riga di comando dentro al contenitore potete modificarli direttamente con il vostro editor preferito nella cartella locale del vostro host ~/**codice/lxp**.

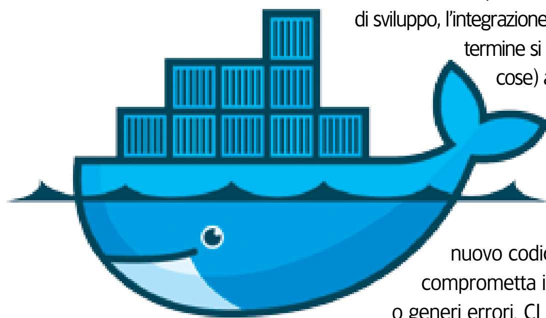
La crescita continua di Docker

Si è parlato molto di Docker e della Docker Inc. (la compagnia che sta dietro al progetto) nell'ultimo anno anche grazie all'annuncio dell'investimento di 40 milioni di dollari da parte di Sequoia, il ben conosciuto VC che ha supporato Google e molti altri nomi familiare nel settore tecnologico. Un importo simile è una grande spinta che dovrebbe permettere a Docker di andare oltre a ciò che è ora, un formato per contenitori, con un sano ecosistema di collaboratori che cresce quotidianamente (basta dare un'occhiata su GitHub cercando Docker per capirlo). Docker è adottato ogni giorno da nuove piattaforme e progetti (Core OS, Apache Mesos) e ha diverse compagnie importanti (eBay, Spotify, Baidu) che lo utilizzano. L'obiettivo di Docker Inc sembra essere quello di rendere il software più adatto ad ambienti di produzione oltre che allargare il bacino di utenti; clustering, scheduling, storage orchestration e networking sono tra le aree di miglioramento. Quando questi miglioramenti saranno disponibili nasceranno sicuramente moltissimi tool di gestione e servizi di supporto commerciale. Questo è il modo in cui Docker punta a guadagnare soldi e chi ha investito in esso spera di vedere un ritorno economico. La scommessa in questo caso è se Docker riuscirà a diventare il nuovo standard per offrire applicazioni "in cloud". Docker rimarrà Open Source, ovviamente, e la comunità continuerà a migliorarlo e svilupparlo in tutti i modi, alcuni di essi inaspettati.



❯ **Ecco la prima parte del nostro job Jenkins, ci mostra le informazioni di accesso di Git**

aprendo la directory **/tmp/registry** che contiene una struttura di cartelle. Ovviamente, in una situazione reale Registry dovrebbe trovarsi in un server dedicato, raggiungibile da tutto il team di sviluppo. Con questo scopo la raccomandazione è quella di nasconderlo dietro a un Web server come Nginx (o Apach). Trovate la documentazione delle funzioni avanzate all'indirizzo **http://bit.ly/DockerRegistryAdvanced**. Ora che abbiamo un modo per condividere i nostri Dockerfile con i membri del nostro staff, dobbiamo soddisfare un requisito dei moderni ambienti di sviluppo, l'integrazione continua (CI). Questo termine si riferisce (tra le altre cose) alla pratica di eseguire test sul nostro codice per assicurarci che quando viene aggiunto del nuovo codice al sistema, non comprometta il progetto o generi errori. CI è un argomento



molto vasto e farne un'analisi approfondita va oltre le possibilità di questo articolo. Per il momento, comunque, assumiamo che l'obiettivo da raggiungere sia quello di eseguire un sistema di CI Open Source, nello specifico Jenkins. Offre una un'ampia gamma di utilizzi ed è supportato da una grande comunità.
In un progetto pre-Docker, questo avrebbe comportato la creazione di un nuovo server, installare JDK (Jenkins è scritto in Java) e quindi scaricare il software Jenkins. Ora, con Docker disponibile, creare un sistema Jenkins basilare è tanto semplice quando digitare questo codice:

```
sudo docker pull jenkins
sudo docker run --name localjenkins -p 8080:8080 -v /var/jenkins_home jenkins
```

Dopo aver scaricato l'immagine di Jenkins dall'Hub di Docker (può metterci un po' di tempo), abbiamo avviato il server Jenkins sulla porta 8080 e aggiunto un volume persistente per i suoi dati.
Da notare che non lo abbiamo collegato a una posizione locale (avremmo potuto farlo facilmente utilizzando la stessa sintassi dell'esempio precedente) ma possiamo esportare dati dal nostro sistema Jenkins (o da qualunque altro contenitore Docker) utilizzando il comando **docker cp**.

Dock continui

In alcune circostanze Jenkins può eseguire interi gruppi di test durante il processo di costruzione di un'applicazione. Alla fine di questo processo, possono nascere alcuni nuovi pacchetti o eseguibili; in alcuni casi vengono eseguite delle macchine virtuali ed eseguiti dei test su di esse per controllare il codice. Non sarebbe una gran cosa se potessimo sfruttare il basso consumo di risorse di Docker per lanciare un contenitore con queste funzioni? O, ancora meglio, far sì che Jenkins importi e agisca all'interno del nostro Registry Docker locale? Certo che sì! Chiudiamo il contenitore con Jenkins (basta premere **CTRL+C** nella finestra). Ci torneremo fra poco. Prima di proseguire dobbiamo anche permettere a Docker di ascoltare comandi remoti oltre alle porte su cui è in ascolto di default. In uno scenario reale dovremmo prestare maggior attenzione alla sicurezza, ma ai fini di questo tutorial ci va bene così. Utilizzando **sudo** modifichiamo il file **/etc/default/docker** (il file si chiama **docker.io** se siete su Ubuntu) aggiungendo questa riga:

```
DOCKER_OPTS="-H 0.0.0.0:4243 -H unix:///var/run/docker.sock"
```

Simuliamo ora il nostro primo check in di codice Ruby on Rails che il nostro team di sviluppo ha scritto utilizzando il nostro container Rails: nel nostro caso è la struttura che abbiamo definito nel primo Dockerfile. Per prima cosa, creiamo una directory locale in cui salvare

Microservizi

Il termine *microservices* (microservizi) si riferisce a un'architettura software che è cresciuta in popolarità negli ultimi tempi. L'idea di base è di rimpiazzare le applicazioni monolitiche che si trovano tra l'interfaccia del client (spesso il browser) e il back-end del database con elementi individuali che fanno attività specifiche più piccole. Questo, in un certo senso, rispecchia l'idea di Unix, molte piccole applicazioni che fanno tutto molto bene. Con le applicazioni monolitiche, le modifiche diventano sempre più difficili mano a mano che l'applicazione cresce fino al punto che anche una minima modifica di un elemento può comportare problemi all'intera app. Scalare può diventare costoso per le applicazioni troppo grandi e dispersive, specialmente nel caso in cui soltanto un elemento ha bisogno di modifiche ma tutto è impacchettato insieme. Con l'approccio dei microservizi, ogni elemento è un servizio separato, permettendo a ognuno di essi di essere sviluppato e distribuito in maniera indipendente. Questi servizi comunicano tra di loro attraverso interfacce ben note. Lo sviluppo a microservizi può essere gestito da team più piccoli, utilizzando qualsiasi tool o linguaggio essi ritengano più adatti alle loro necessità. Sono molto più elastici e molto più semplici da sostituire, evitando problemi di ereditarietà. I detrattori di questo approccio lo definiscono 'SOA hipster' (*service oriented architecture* (architettura orientata ai servizi)) e asseriscono che aumenti il livello di complessità nelle infrastrutture. A parte tutto, non è difficile capire perché Docker abbia stimolato l'immaginazione dei fautori dei microservizi e sia stato adottato rapidamente in questo tipo di progetti, visto che sembra progettato appositamente per soddisfare queste necessità.

il codice, assumendo di essere nella nostra directory **home**, un semplice **mkdir codice** risolverà il problema. Poi, riutilizziamo la nostra immagine railstest di prima:

```
sudo docker run -p 3000:3000 --name sviluppoirails -v ~/codice:/codice -t -i railstest /bin/bash
```

Questo ci porta dritti dentro al prompt all'interno di un nuovo contenitore, con la nostra cartella 'codice' condivisa all'indirizzo **/codice**. Copiamo qui il nostro codice sorgente e verifichiamo che sia corretto. Non preoccupatevi troppo di questa fase... non è un tutorial su Rails!

Fare il check in della nostra app

Abbiamo un'altra piccola attività da fare prima di utilizzare Jenkins, creare un repository Git su cui utilizzarlo. In un contesto reale, questo si troverebbe in un altro server (o contenitore Docker) o meglio ancora ospitato in rete da qualche servizio tipo GitHub. Assicurandovi di essere nel contenitore giusto, eseguite questi comandi (sostituendo ovviamente l'indirizzo email e il nome):

```
cd /codice
cp -r /root/lxp .
cd lxp
git init
touch git-daemon-export-ok
mkdir docker
git add .
git config --global user.email "indirizzo@dominio.it"
git config --global user.name "Vostro nome"
git commit -m "Check in iniziale"
```

Questi comandi creano un nuovo repository Git nel nostro disco locale contenente l'intera struttura dell'applicazione Rails. Il nostro obiettivo è quello di far sì che Jenkins legga questo repository, crei una nuova immagine Docker dal nostro Dockerfile, la avvii e controlli il nostro codice mentre lo fa. Come prima, in un contesto reale probabilmente avremmo i Dockerfile e il codice delle app in repository separati, ma per praticità, creiamo un nuovo Dockerfile all'interno della sottodirectory docker del nostro repository:

```
FROM railstest
ENV HOME /root
RUN cd $HOME; rm -fr lxp
RUN git clone git://<ip della vostra macchina>/lxp
```

Inviamo questa modifica con **git add .** e **git commit -m "aggiunto il Dockerfile"**. Lanciamo ora un semplice server Git per permettere a questi file di essere letti da un nuovo container Docker. Aprite un terminale, fate **cd** nella directory **~/codice/lxp** e lanciate quanto segue:

```
sudo git daemon --reuseaddr --base-path=/home/<il vostro nome>/codice --export-all
```

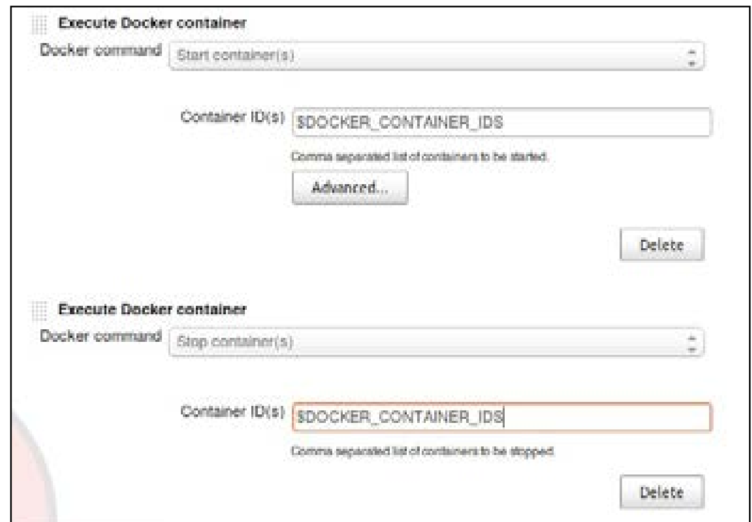
Lasciate questa finestra aperta da qua in poi.

Jenkins, creami un nuovo contenitore Docker

In una finestra separata, avviate nuovamente Jenkins, se volete riutilizzare lo stesso nome per il contenitore dove prima eliminare il precedente:

```
sudo docker rm localjenkins
sudo docker run --name localjenkins -p 8080:8080 -v /var/jenkins_home -v ~/codice:/codice jenkins
```

Aprendo un browser nel computer locale sarà ora possibile connettersi a **http://127.0.0.1:8080** e visualizzare la pagina Jenkins. Per prima cosa bisogna installare il plug-in Git per far sì che Jenkins riesca a leggere il nostro codice. Clicchiamo sul link **Manage Jenkins** nella parte sinistra dello schermo, e poi su **Manage Plugins**, che è la quarta opzione partendo dall'alto. A questo punto andiamo nel tab **Available**, e nella casella di ricerca scriviamo **Git Plugin**. In questo modo possiamo ridurre l'elenco degli svariati plug-in disponibili per Jenkins. Selezioniamo la checkbox a sinistra di **Git Plugin** e clicchiamo su **Install without restart**. Completata l'installazione, torniamo nei **Plugin**, di nuovo **Available** e cerchiamo **docker**. Quello che vogliamo è **Docker Build Step Plugin**. Come prima, selezioniamo e installiamo senza riavviare. Stessa cosa per un terzo plug-in **Token Macro Plugin**. Una volta completato, torniamo al link **Manage Jenkins** e scegliamo **configure system**. Dobbiamo configurare il nostro plugin Docker e assicurarci di poter comunicare con il nostro host Docker (il nostro desktop in questo caso). Scrolliamo in basso in questa pagina e inseriamo in Docker Url la stringa **http://<ip della vostra macchina>:4243**. Possiamo testare la connessione con l'apposito pulsante e se tutto va a buon fine comparirà un messaggio **Connected to** seguito dall'indirizzo inserito. Clicchiamo **Save**. Ora possiamo cliccare sul link **create new jobs**, chiamiamo il nostro **job testdocker**, scegliamo l'opzione **free-style software project** e clicchiamo **OK**. Per l'opzione **Source Code Management** scegliamo **Git** (grazie al nostro plug-in) e inseriamo come URL **file:///codice/lxp** (non abbiamo bisogno di credenziali di accesso). Scendiamo in fondo alla pagina e scegliamo **Add build step**, cliccando su **Execute Docker Container** nella tendina. In **Docker command** scegliamo **Create Image**. L'opzione di default per la cartella dei contenuti è a posto, dobbiamo solo modificare il tag aggiungendo **rails_** davanti a esso. Aggiungiamo un secondo step, questa volta scegliamo **Create container**, come **image name** mettiamo il tag del passo precedente. L'hostname può essere qualsiasi. Aggiungiamo un terzo passo, questa volta **Start container(s)** e mettiamo come ID **$DOCKER_CONTAINER_IDS** (è una variabile di ambiente del plug-in). Per concludere aggiungiamo un quarto step **Stop container(s)**, mettiamo nuovamente **$DOCKER_CONTAINER_IDS** come ID nel campo di testo. Ora possiamo cliccare su **Save** e scegliere l'opzione **Build Now** a sinistra. Jenkins andrà a controllare il nostro file Docker, costruire un'immagine, eseguire l'immagine e una volta confermato che tutto è andato a buon fine, la chiuderà. Potete controllare lo stato del lavoro (rosso è cattivo, blu è buono!) e guardare l'output della console per sapere cosa sta facendo. Il comando **sudo docker images** ci mostra le immagini rails_* create in questo modo. Questo semplice processo può essere utilizzato come base per un sistema CI che utilizza Docker e può essere espanso coinvolgendo sempre più codice e maggiori test sulle applicazioni. Buon divertimento! LXP



› **La seconda parte del nostro job Jenkins ci mostra come possiamo interagire con Docker e le variabili di ambiente**

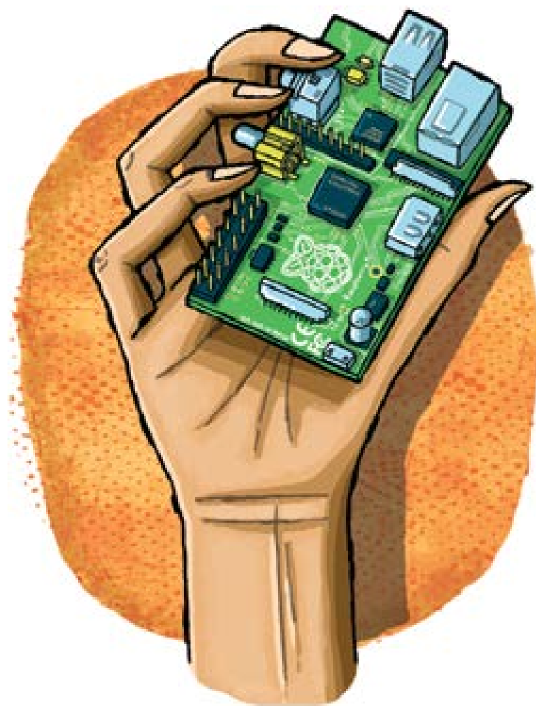
Tip

Potete eseguire i job di Jenkins in qualsiasi momento durante la loro creazione, basta salvarli e premere **Build Now**. Sperimentate con le varie opzioni per vedere che errori appaiono!

Creare pagine XHTML

Con una pila LEMP installata lo staff di Linux Pro vi spiegherà come creare siti dinamici e farli funzionare anche con un Raspberry Pi





Abbiamo già visto come installare e configurare un server **Nginx** in un articolo precedente e ora possiamo far progredire il nostro sito iniziando a introdurre HTML (*Hyper Text Markup Language*) e CSS (*Cascading Style Sheets*). HTML verrà utilizzato per creare il contenuto della pagina Web e CSS per migliorare l'aspetto del sito. Parleremo più dettagliatamente di HTML e CSS durante il tutorial e parleremo inoltre di un cugino dell'HTML, lo standard **XHTML**. Grazie a questi strumenti sarete in grado di progettare e costruire pagine Web con un aspetto molto gradevole che soddisfa gli standard Web e garantendo piena compatibilità con tutti i browser. Noi stiamo utilizzando un Raspberry Pi con 512 MB di RAM con Debian Wheezy 7.5 e il server Nginx installato e in esecuzione. Abbiamo configurato come root di Nginx la directory **/var/www** e ovviamente è necessario avere i privilegi per poter modificare il contenuto di questa directory per poter procedere. Come detto prima HTML è l'acronimo di *Hyper Text Markup Language* (linguaggio a marcatori per ipertesti) e rappresenta lo standard gestito dai browser Web come Firefox o Safari per visualizzare testi, immagini e collegamenti con un certo grado di formattazione. L'HTML è stato sviluppato inizialmente per mostrare documenti. Molto di ciò che esisteva all'inizio in Internet era legato all'ambiente universitario da dove è partito gran parte dello sviluppo iniziale. La possibilità di collegare una parola al lavoro a cui si riferisce è un aspetto fondamentale: in questo modo chi sta leggendo un documento può facilmente passare all'articolo accademico a cui fa riferimento. Molta gente ritiene che Tim Berners-Lee sia il padre di Internet ma la realtà è che ha sviluppato la rete che ha reso Internet utilizzabile. Internet permette ai computer di parlare tra di loro ed è stato sviluppato dall'esercito americano e dalle università americane negli anni '60. Tim Berners-Lee, un laureato di Oxford, ha sviluppato quello che conosciamo ora come Internet nel 1990 e ha definito le specifiche per HTML, URI (*Universal Resource Identifier*, che permette di localizzare i documenti nella rete) e HTTP (*Hyper Text Transfer Protocol*, il linguaggio utilizzato dai browser Web per parlare con i server). Ancora oggi è il direttore del W3C (**World Wide Web Consortium**), il consorzio che gestisce e sviluppa gli standard per i protocolli Web.

L'HTML

Utilizziamo l'HTML per scrivere documenti di testo che diventano poi pagine Web. Una pagina Web è semplicemente un file di testo; utilizziamo il codice HTML per gestire il testo in modo che venga visualizzato in grassetto o che diventi un collegamento. Utilizzare i "tag" HTML ci permette di definire elementi all'interno della pagina che vengono trattati in modo particolare dai browser. Ogni elemento ha un tag di apertura e uno di chiusura che definisce l'inizio e la fine di quell'elemento. Lo standard HTML determina la sintassi e quali tag vanno utilizzati per la chiusura: proseguendo nel tutorial andremo a vedere anche quali standard possiamo implementare. Una pagina Web normalmente ha come estensione '.html', la pagina di benvenuto di un sito Web solitamente viene chiamata index.html. Questa pagina è quella di default che viene chiamata se nell'URI non è specificata alcuna pagina. Per esempio **www.esempio.it** non specifica il documento, di conseguenza verrà ritornata index.html; se invece utilizziamo l'URI **www.esempio.it/informazioni.html** verrà ritornata la pagina specificata. Il file index.html può essere creato con qualsiasi editor di testo. Comunque, in base a quanto complessa volete che sia la pagina, un editor che gestisca l'HTML può essere molto di aiuto. Su Raspberry Pi possiamo utilizzare **nano** o anche **vi** per creare e modificare il documento da riga di comando e **LeafPad** se preferiamo una GUI. Potremmo anche valutare di installare **Geany**, che è un editor di testo e un IDE (*Integrated Design Environment*, ambiente di sviluppo integrato) Open Source che può aiutarci a scrivere HTML e codice in altri linguaggi come C o Python. La colorazione del codice può rendere la pagina molto più facile da leggere durante le modifiche e in Geany, per esempio, quando viene aperto un tag viene creato in automatico anche quello corrispondente di chiusura. Per installare Geany su Raspberry Pi basta digitare:

```
sudo apt-get update
sudo apt-get install geany
```

Capire i tag

Come abbiamo visto in passato, è possibile creare una pagina Web utilizzando soltanto del testo senza alcun tag HTML. In questo caso il testo viene visualizzato senza modifiche grafiche, blando, senza evidenziare le informazioni ne suscitare attrattiva sui lettori. Aggiungeremo dei tag HTML per aumentare l'interesse e la leggibilità della nostra pagina, iniziamo aggiungendo il DOCTYPE.

» **<DOCTYPE>** Questo tag serve soltanto a scopo informativo,

Validare l'HTML

Tutto il codice HTML e XHTML può essere validato con facilità da W3C direttamente online, basta andare sul sito ufficiale: **http://validator.w3.org**. Potete incollare del codice dentro al validatore, caricare un file HTML o validare direttamente una pagina online accessibile in Internet. Tutto il codice validato con successo può utilizzare il logo W3C che indica quale standard rispetta la vostra pagina. Questo può aiutarvi anche a imparare la sintassi corretta per rispettare gli standard dato che quando vengono rilevati dei problemi viene spiegato anche a cosa è dovuto l'errore. Infine, scrivere del codice che rispetta uno standard vi garantisce che la vostra pagina venga visualizzata come volete in tutti i browser.

non concorre all'aspetto della pagina. Il tag è utilizzato per specificare lo standard HTML in cui è scritto il documento in modo da far capire al browser in che modo interpretare il documento. Questo tag può essere omesso ma rischiamo che il browser scelga di interpretare il nostro codice con uno standard diverso da quello da noi scelto. Chiedere ai software di indovinare non è mai una grande idea. Includere questo tag ci permetterà anche di far validare il nostro codice in quello standard in modo da essere sicuri di aver scritto la pagina in modo corretto senza aver dimenticato attributi o tag di chiusura. Potete validare il vostro codice a questo indirizzo: **http://validator.w3.org**. Per questo tutorial utilizzeremo il **DOCTYPE** dell'XHTML, in questo modo:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

I più esperti tra di voi avranno notato che abbiamo utilizzato anche **XML** per impostare la codifica dei caratteri da utilizzare, in questo caso **UTF-8**. Fa parte dello standard XHTML e può essere impostato nel documento o direttamente nel server Nginx con la direttiva: "**charset utf-8;**". Perché XHTML invece di HTML? XHTML sta per *eXtensibile HyperText Markup Language* ed è quasi identico all'HTML 4.01 a parte il fatto che è uno standard più rigoroso e più chiaro. XHTML ridefinisce HTML come un'applicazione XML supportata da tutti i principali browser. XHTML offre uno standard migliore per gli sviluppatori e i browser rendendo obbligatori molti elementi opzionali dell'HTML come i tag <html>, <head>, <title> e <body>. I tag devono essere scritti in minuscolo; fatta eccezione per il tag DOCTYPE. Per rendere il codice più leggibile ogni elemento deve essere chiuso. In HTML iniziare un nuovo paragrafo con <p> fa sì che quello precedente venga automaticamente chiuso se era ancora aperto.
Per correttezza però il tag andrebbe chiuso in modo esplicito con </p>. In XHTML la chiusura dei tag è obbligatoria risolvendo molti problemi derivanti da una programmazione disattenta. Alcuni tag possono chiudere loro stessi: vale per gli elementi vuoti come la riga orizzontale <hr> o l'invio
. Per farli chiudere è necessario scriverli in questo modo
. Questa sintassi apre e chiude correttamente il tag secondo lo standard XHTML. Dato che l'XHTML è lo standard HTML attuale per W3C lo implementeremo nel nostro sito. HTML 5 è in lavorazione e al momento è una bozza di standard e non è supportato da tutti i browser.

» <html> Il tag <html> definisce la radice del nostro documento HTML ed è obbligatorio in XHTML come l'attributo xmlns, che specifica il namespace XML da utilizzare nel documento. Il tag <html> funziona da contenitore per tutti gli elementi della pagina. Un esempio di tag <html> dichiarato correttamente in XHTML potrebbe essere: <html xmlns="http://www.w3.org/1999/xhtml">. Come detto in precedenza tutti gli elementi, incluso questo tag, devono essere in minuscolo e gli attributi devono essere tra virgolette.

» <head> Il tag <head> è un elemento contenitore in HTML e contiene i tag per i metadati che descrivono il documento HTML



› Possiamo utilizzare il validatore di www.W3.org per controllare che il nostro HTML rispetti lo standard

stesso. Per la maggior parte questi dati non sono visibili agli utenti: l'eccezione più ovvia è l'elemento <title> che può essere utilizzato per mostrare il titolo della pagina nel tab del browser. Oltre all'elemento <title>, il <body> di solito contiene degli elementi <meta> che danno informazioni relative alla pagina ai motori di ricerca e gli elementi <link> permettono di importare fogli di stile. In alternativa si può utilizzare l'elemento <style> per definire stili all'interno della pagina stessa. Per concludere, l'elemento <script> definisce gli script che devono essere eseguiti nella pagina. Vediamo un possibile tag head:

```
<head>
        <title>Linux Pro - Fondamenti di HTML</title> <link
rel="stylesheet" href="/stylesheets/screen.css" type="text/css" />
        <meta name="Author" content="Linux Pro Staff" />
        <meta name="Description" content="Tutorial -
Fondamenti di HTML" />
        <script type="text/javascript">
                window.onload = function() { alert("Il
Raspberri Pi è fantastico!");
        };
        </script>
</head>
```

» <body> L'elemento <body> funziona da contenitore per gli elementi che verranno visualizzati dal browser. La maggior parte del tempo che passeremo lavorando sulle pagine Web sarà all'interno dell'elemento <body>. Solitamente il primo tag all'interno del body è l'elemento descrittivo <h1>: abbiamo tag da <h1> fino a <h6> per definire le intestazioni. All'interno di un documento dovrebbe comparire solo un tag <h1> perché è l'elemento principale utilizzato dai motori di ricerca per determinare di cosa parla la pagina. All'interno del body possiamo utilizzare altri elementi, come <div> per spezzare la pagina in porzioni logiche che possono essere gestite separatamente. Ogni pagina Web contiene un singolo elemento <body> ma il <body> può contenere molti elementi <div>. Utilizzando gli attributi degli elementi con gli stili CSS possiamo definire il colore del testo, il font da usare, le dimensioni, così come riordinare gli elementi nella pagina. Per ogni elemento possiamo utilizzare l'attributo ID e l'attributo CLASS. L'ID è utilizzato per identificare in modo univoco un elemento nella pagina; CLASS si usa per raggruppare gli elementi che necessitano di uno stile

simile. Se vogliamo inserire un blocco di testo che funga da introduzione seguito dal testo principale possiamo considerare il codice HTML che segue, il quale utilizza degli stili embeddati:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
        <head>
                <title>Linux Pro</title>
                <link rel="stylesheet" href="stylesheets/
screen.css" type="text/css" />
                <meta name="Author" content="Linux Pro
Staff" />
                <meta name="Description"
content="Tutorial HTML" />
                <script type="text/javascript">
                        window.onload = function() {
alert("Il Raspberri Pi è fantastico!");
                };
                </script>
                <style type="text/css">
                        body {color:red}
                        #principale {color:darkslategray;
font-size:x-large}
                        #principale::first-letter {color:red;
font-size:xx-large}
                        #contenuto {color:lightslategray;
font-size:medium}
                        .tutorial {text-indent:50px}
                </style>
        </head>
        <body>
                <h1>Fondamenti di HTML</h1>
                <div id="principale" class="tutorial">
                        <p>Lezione sull'HTML e sull'uso
dei tag</p>
                </div>
                <div id="contenuto" class="tutorial">
                        <p>L'HTML &egrave; stato
sviluppato da Tim Berners-Lee. <br /> Tim si &egrave; laureato
all'universit&agrave; di Oxford.</p>
                </div>
        </body>
</html>
```

Gli elementi di stile inclusi nella pagina colorano il testo dentro all'elemento <body> di rosso; l'elemento con ID 'principale' sarà di colore **darkslategray**, un grigio scuro; l'elemento con ID 'contenuto' invece sarà di un grigio più chiaro. Gli stili controllano anche le dimensioni dei font. Per movimentare il tutto abbiamo indentato la prima riga di ogni elemento con classe tutorial di 50 pixel e la prima lettera dell'elemento con ID 'principale' sarà rossa

› La nostra pagina XHTML iniziale: un punto di partenza a cui aggiungere stili

Fondamenti di HTML

- Home
- Ricerca

Benvenuti nel tutorial sull'HTML e CSS3 di Linux Pro

e più grande rispetto al resto del testo. Nel CSS ci si riferisce agli id utilizzando il simbolo # mentre il punto si riferisce alla classe degli elementi. Gli elementi HTML possono essere identificati dal tag come possiamo vedere dall'uso di body nel CSS.

» Spazi bianchi l'HTML ci permette di inserire un singolo spazio nel testo di una pagina e tutti gli spazi bianchi aggiuntivi come spazi o tab vengono ignorati. Possiamo aggiungere spazi multipli tra le parole di un documento ma soltanto uno spazio verrà aggiunto e gli a capo verranno ignorati. Per aggiungere una singola riga possiamo utilizzare l'elemento autoconclusivo
 oppure possiamo utilizzare <p> dove vogliamo iniziare un nuovo paragrafo e verranno aggiunte due righe. Ogni elemento <div> aggiungerà una nuova riga. Un <div> può contenere uno o più elementi <p> ma il paragrafo, <p>, non può contenere elementi <div> secondo lo standard XHTML.

» Caratteri speciali Lo standard HTML prevede una codifica particolare per i caratteri speciali: anche le lettere accentate sono considerate caratteri speciali e quindi richiedono una sintassi propria. Nel codice scritto poco fa potete notare queste stringhe "è" e "à" che corrispondono rispettivamente alla à e a è: come potete notare questa sintassi è rappresentata da una & seguita dalla lettera interessata (ovviamente rispettando maiuscole e minuscole: è diventa è, mentre È diventa È) seguita a sua volta da "grave" o "acute" a seconda che l'accento sia grave o acuto, il tutto concluso con un punto e virgola.

CSS3

Abbiamo già visto un po' di stili nella pagina XHTML su cui stiamo lavorando; piazzare lo stile all'interno dell'elemento <head> non ci permette di usare lo stesso stile in tutte le nostre pagine. Certo potremmo copiare e incollare le informazioni dello stile in ogni pagina ma sarebbe difficile mantenere il codice dato che ogni modifica dovrebbe essere replicata in tutte le nostre pagine.
Per utilizzare gli stili come sono stati pensati dobbiamo utilizzare l'elemento <link> nel tag <head> per puntare a un foglio di stile esterno che serva tutte le pagine. Utilizzeremo CSS3 che è basato sul core di CSS2.1 con dei moduli aggiuntivi. Per prima cosa, creiamo una pagina XHTML come base per il nostro progetto e aggiungiamo lo stile a questa e ad altre pagine:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
        <head>
                <title>Linux Pro</title>
                <meta name="Author" content="Linux Pro
Staff" />
                <meta name="Description"
content="Tutorial HTML" />
        </head>
        <body>
        <div id="pagina">
                <div id="header" class="frame">
                        <h1>Fondamenti di HTML</h1>
                </div>
                <div id="menu" class="frame">
                        <ul>
                                <li><a id="home" href="/
index.html">Home</a></li>
                                <li><a id="ricerca" href="/
ricerca.html">Ricerca</a></li>
                        </ul>
```

```
                </div>
                <div id="contenuto" class="frame">
                        Benvenuti nel tutorial sull'HTML
e CSS3 di Linux Pro </div>
                </div>
        </body>
</html>
```

Questo codice non produce una pagina dall'aspetto molto curato, ma ci dà diverse informazioni su cui possiamo lavorare per ottenere un aspetto migliore. Per semplicità creeremo un foglio di stile nella stessa directory del file index, la **/var/www** sul Raspberry Pi. Utilizziamo Geany o un altro editor di testo per creare un file chiamato **/var/www/stile.css**. Puntare al foglio di stile dalla nostra pagina index oltre che da tutte le altre, come per esempio la pagina di ricerca, è semplicissimo, basta aggiungere all'interno dell'elemento <head> quanto segue:

```
<link rel="stylesheet" href="/stile.css" />
```

Il nostro file CSS comincerà con la definizione del set di caratteri da utilizzare:

```
@charset 'utf-8';
```

Se guardiamo il codice XHTML abbiamo un elemento <div> con id pagina che è il contenitore principale di tutti gli altri elementi. Ora vediamo di dare uno stile a questo elemento:

```
#pagina { width:700px; margin:auto; border: 1px solid
darkslategray; border-radius: 5px; box-shadow: 4px 4px 4px black;
}
```

Quando si definiscono gli stili, ogni stile ha un nome seguito dai due punti (**:**) e da un valore: ogni attributo termina con un punto e virgola (**;**). Le istruzioni possono essere scritte su più righe per migliorare la leggibilità. Aggiungere questo stile e aggiornare il browser risulterà in una pagina con un blocco centrale con gli angoli arrotondati e un po' di ombra.

Il prossimo elemento nella nostra pagina XHTML è il blocco con id header. Questo elemento funziona come banner principale e noi imposteremo il colore di sfondo e l'altezza. Aggiungendo quanto segue al CSS impatteremo soltanto sull'elemento con id header:

```
#header { background-color: slategray; color: white; height: 55px;
}
```

Abbiamo incluso la classe frame in ogni elemento <div> all'interno del div della nostra pagina. La classe frame può essere utilizzata per dare uno stile a tutti i blocchi contemporaneamente

```
.frame { width: 700px; margin: 0px; padding 0px; background-
color: white; }
```

Abbiamo impostato il colore di sfondo con l'istruzione **background-color** sia all'ID sia alla classe: l'id ha la meglio perché è un tag più specifico e quindi l'elemento con id header ha come colore di sfondo **slategray**. Per formattare l'elemento h1 nel nostro elemento header possiamo utilizzare lo stile che segue, che coinvolge soltanto gli elementi h. Se ricordate abbiamo detto che bisognerebbe sempre avere soltanto un h1 all'interno di una pagina:

```
h1 { text-align: center; text-shadow: 4px 4px 4px black; font-size:
300%; }
```

Il prossimo elemento della pagina è il contenitore <div> con ID menu. Questo blocco contiene i link per la navigazione all'interno del nostro sito. I link sono racchiusi dentro una lista non ordinata e tutti questi elementi hanno bisogno di un po' di stile:

```
)#menu { font-family: sans-serif; background-color: black; height:
22px; line-height: 22px; position: relative; }
#menu ul { list-style-type: none; margin: 0; padding: 0; }
#menu li { display: inline; float: left; position: relative; padding: 0
5px; }
#menu a:link, #menu a:visited { background-color: black; color:
white; display: block; font-weight: normal; padding: 0 5px; text-
decoration: none; }
#menu li a:hover, #menu li a:active, #menu li.selected a:link,
#menu li.selected a:visited { background-color: white; color: black;
}
#menu li {
   display: inline; float: left; position: relative; padding: 0 5px;
   }

#menu a:link, #menu a:visited {
   background-color: black; color: white;
   display: block; font-weight: normal;
   padding: 0 5px; text-decoration: none;
}
#menu li a:hover, #menu li a:active,
#menu li.selected a:link, #menu li.selected a:visited {
   background-color: white; color: black;
```

Abbiamo molto codice qui: coloriamo di nero lo sfondo della barra del menu e il testo bianco; disponiamo i link in orizzontale e scambiamo i colori quando ci si passa sopra con il mouse. L'effetto è abbastanza piacevole e non è molto difficile da ottenere. Per finire abbiamo il blocco con i contenuti:

```
#contenuto { padding: 25px 50px; width: 600px; font-size:
150%; }
```

Impostiamo i margini interni (padding) a 25 pixel per sopra e sotto, e 50 pixel per destra e sinistra. Impostiamo la larghezza dell'elemento a 600 pixel dato che dobbiamo farlo stare dentro i 700 pixel. Finalmente la nostra pagina ha un aspetto migliore ma non abbiamo ancora una pagina di ricerca ed è facile utilizzare la pagina index come template per le altre pagine e lo stile persiste in tutte le pagine. La pagina di ricerca può essere una semplice copia dell'index modificando gli elementi <title> e l'<h1>. LXP



› Ora con un po' di stile grazie al file che abbiamo creato in /var/www

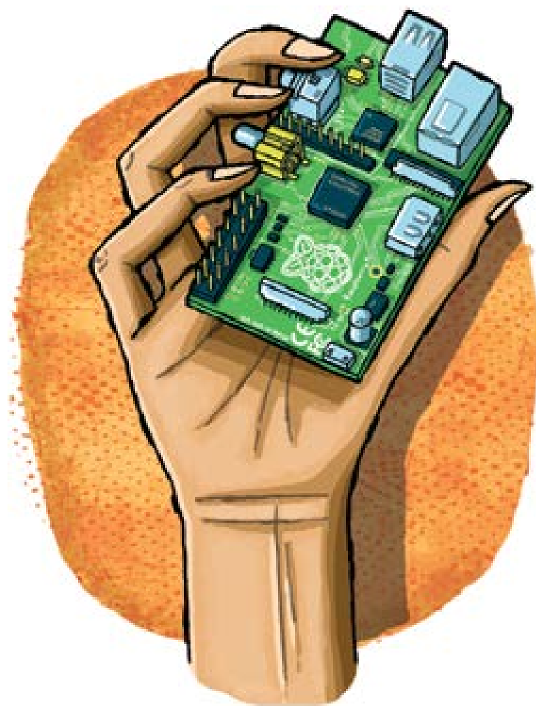
CSS padding e margini

Quando si aggiunge del padding o dei margini a un elemento è necessario capire che il padding è la distanza degli elementi dentro al blocco dal bordo del blocco stesso, mentre i margini sono la distanza dal bordo dell'elemento dagli altri elementi della pagina. Entrambi hanno quattro attributi principali: i valori **top** (sopra), **right** (destra), **bottom** (sotto) e **left** (sinistra). Se vengono passati tutti e 4 i valori, vengono letti in questo ordine. Se ne passate soltanto due il primo vale per sopra e sotto, il secondo valore invece per destra e sinistra. Se viene passato un valore solo tutti e quattro gli attributi utilizzeranno quel valore.

Collegarsi a un database

Ecco come completare perfettamente la creazione di siti Web dinamici usando Raspberry Pi e una connessione MySQL





In questa ultima parte della serie su Nginx/LEMP costruirete una soluzione per lavorare con un database MySQL e creare un semplice database di contatti. Una volta che il database sarà in esecuzione e conoscerete alcune tecniche basilari per le query SQL, integrerete questo pezzo finale del puzzle nel vostro sito in PHP, che vi permetterà di eseguire ricerche remote nel database utilizzando form HTML nel server Web Nginx.

Creare un database

Se non avete ancora installato il vostro MySQL, fate riferimento alle puntate precedenti. Il server database manterrà l'archivio persistente di tutto il sistema. Il database che creerete fungerà da contesto per contenere e ricercare le tabelle, che a loro volta contengono i vostri dati. Prima di procedere potete fare alcuni controlli per verificare che il server del database sia attivo e funzionante. La cosa più semplice è usare la riga di comando:

```
sudo service mysql status
```

L'output risultante contiene tutte le informazioni riguardo l'attività del vostro database. Dovete anche assicurarvi che il server possa essere acceduto solamente da localhost: in altre parole il server Web, che gira sulla medesima macchina, dovrebbe poter connettersi al database, ma non client remoti. Potete visualizzarlo usando i comandi Linux **ss o show sockets**:

```
ss -lt | grep mysql
```

L'output di **ss** vi dovrebbe mostrare che il servizio di MySQL è in ascolto solamente su 127.0.0.1, il localhost, e non l'interfaccia esterna del Pi. La porta in uso dal servizio è la 3306. La porta rappresenta solamente l'indirizzo del servizio: saprete di certo che il server Web di Nginx ascolta sulla porta 80, e la porta standard di MySQL è la 3306. Per vedere le mappature comuni delle porte con i relativi nomi di servizio potete leggere il file **/etc/services**. Su Pi questo file ha più di 600 righe, tuttavia potete trovare l'informazione che cercate con la shell:

```
grep mysql /etc/services
```

Potete restare loggati in Raspberry Pi come utente standard e da LXTerminal autenticarvi al servizio MySQL con il client a riga di comando *mysql*. Dal momento che effettuate il login in MySQL indipendentemente, non necessitate di privilegi di root in Linux. Tuttavia non siate confusi, vi registrerete in MySQL con l'account root di MySQL stesso. Attualmente c'è un solo account utente nel server database e ha pieni diritti a tutte le risorse di MySQL allo stesso modo in cui un account root ha pieni diritti su un sistema Linux. Utilizzando il client mysql potete entrare con il seguente comando:

```
mysql -u root -p
```

Vi verrà chiesta una password e, dopo l'autenticazione, vi sarà presentato il prompt di MySQL. Potete concentrarvi ora sull'importante compito di creare un database.
Il database è dove sono contenute le tabelle. Sotto molti aspetti quest'ultimo è come una cartella, se confrontato con un normale filesystem. Non dimenticate, nello scrivere istruzioni SQL, di chiuderle con un punto e virgola. Comincerete elencando i database attuali, quindi creando il nuovo database ed elencandoli nuovamente: siete pronti?

```
SHOW DATABASES;
CREATE DATABASE contact;
SHOW DATABASES;
USE contact;
```

Potete notare che avete iniziato con l'esistenza dei soli database standard di sistema e il nuovo database viene quindi creato ed elencato nell'output (contact). Per utilizzare quest'ultimo database utilizzate il comando **use**, analogamente a quanto fate con il comando **cd** nel filesystem. In questo modo potete far riferimento alle tabelle che create con il semplice nome ed evitare la necessità di includere anche il nome del database nel riferimento. Dal momento che avete un database potete ora creare una tabella. La tabella conterrà i dati che aggiungerete per questa semplice applicazione; consiste in una serie di colonne che manterranno l'id utente, il nome e il cognome. Per creare la tabella definite tali colonne con il relativo tipo, oltre ad alcune proprietà aggiuntive per la tabella. Utilizzerete l'istruzione **CREATE TABLE** nel linguaggio SQL per questo primo passo:

```
USE contact;
CREATE TABLE users (
```

Tip

Le istruzioni MySQL possono essere divise in tre tipi principali: DML (*Data Manipulation Language*) che costituisce la maggior parte dei normali insert e select; DDL (*Data Definition Language*) dove create e cancellate oggetti come tabelle, e DCL (*Data Control Language*) con il quale controllate i permessi.

Web service in un Pi

Avete creato qualcosa di straordinario e potente sul piccolo grande Raspberry Pi. Normalmente si parla dei dispositivi embedded come qualcosa di veramente speciale e costoso: sfruttando l a semplicità di Raspberry Pi avete dimostrato come creare un Web service integrato facilmente agganciabile a qualsiasi cosa. Immaginatelo a una fiera dove portate il vostro database di prodotti e lo interfacciate con gli astanti. Il knowhow tecnologico che avete acquisito non si limita a Pi e può essere applicato a qualsiasi distribuzione Linux. Se vi interessa potreste non volervi fermare qui: è solo la prima meta di un viaggio eccitante. **Oracle** mantiene una fornitissima documentazione su MySQL: **http://dev.mysql.com/doc** ed esistono utili risorse su PHP, CSS e HTML, la prima delle quali è **W3Schools**: **www.w3schools.com**.

```
uid INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
fname varchar(50) NULL,
lname varchar(50) NOT NULL);
SHOW TABLES;
```

Una volta creata la tabella potete verificarla dall'output di **SHOW TABLES**. Andando avanti potete usare il comando **DESCRIBE** per dettagliare le colonne create e avere ulteriori informazioni con **SHOW CREATE TABLE**:

```
DESCRIBE users;
SHOW CREATE TABLE users
```

Passiamo ai dettagli riguardo la creazione della tabella. L'avete battezzata users. La definizione, tuttavia, è racchiusa dalle parentesi e consiste in una singola riga di codice MySQL finché non chiudete la parentesi stessa, e alla fine della riga c'è il solito punto e virgola. Per migliorare la leggibilità è possibile spezzarla su più righe così da dare una riga a ogni colonna. Per prima cosa create una colonna uid e accettate un tipo di dato INT (ovvero intero). Impostarlo a **UNSIGNED** indica che accetterete solamente valori positivi e non negativi. Per archiviare questo valore vengono usati 4 byte, permettendo valori da 0 a un enorme 4.294.967.295. Forse un po' troppo per le vostre necessità (avrete mai 4 miliardi di utenti?) ma certamente sufficiente per accettare gli ID sul lungo periodo. **NOT NULL** indica esattamente quello che pensate, assicurando che un valore deva essere per forza presente in questa colonna. Potete, comunque, rendere il processo leggermente più facile con l'opzione **auto_increment**, assicurandovi che i valori vengano aggiunti automaticamente se non forniti. In più tale colonna è stata definita come 'primary key', forzando l'unicità dell'uid per ogni utente nella tabella e offrendo un modo facile per identificare ogni elemento. Potete vedere dal codice che **fname** e **lname** sono molto più semplici nelle loro definizioni. I tipi di entrambe le colonne sono varchar(50), il che permette di inserire fino a 50 caratteri ma si restringe automaticamente in caso siano meno. Nell'esempio per il campo **fname**, inoltre, è stato ammesso il valore nullo.

Inserire i dati nella tabella

Avere una tabella senza dati è un po' come avere una fetta biscottata senza marmellata: le due cose vanno davvero a braccetto. Aggiungete quindi alcuni utenti alla tabella con l'istruzione **MySQL INSERT**. Per prima cosa aggiungete una singola riga, quindi più righe:

```
USE contact;
INSERT INTO users ( fname, lname) VALUES ('Paolo', 'Rossi');
SELECT * FROM users;
```

Iniziate assicurandovi che il contesto del database sia corretto con l'istruzione USE. Anche se siete già nel database è considerato corretto e non genererà alcun errore, costituisce dunque un ottimo sistema di garanzia. L'istruzione **INSERT** aggiunge la riga alla tabella, indicando quali elementi sono forniti tra le prime parentesi e i rispettivi valori nelle seconde. È stata deliberatamente omessa la colonna uid, dal momento che potete lasciarne la compilazione a **AUTO_INCREMENT**. I dati vengono in seguito recuperati con l'istruzione **SELECT**. **INSERT** è simile a una scrittura e SELECT a una lettura. Dalle informazioni risultanti vedete che Paolo ha uid 1. L'uso dell'asterisco all'interno dell'istruzione dice di recuperare tutte le colonne coinvolte. Ora che sapete inserire righe singole, potete vedere come inserirne più di una alla volta. La sintassi è simile ma avrete una lista separata da virgola di valori raggruppati con parentesi come nell'esempio precedente:

```
INSERT INTO users (fname, lname) VALUES ('Giorgio', 'Bianchi'), ('Giuseppe', 'Verdi'), ('Giacomo', 'Virati');
```

Migliorare la SELECT

Con soli tre elementi i risultati non sono difficili da gestire, ma non è un ambiente scalabile mano a mano che più utenti compaiono nella tabella. Per aiutarvi a ridurre le righe ritornate potete usare la clausola **WHERE** nell'istruzione **SELECT**. In questo modo potete mostrare utenti con un dato ID: **WHERE uid = 1** o con un dato cognome: **WHERE lname = 'Rossi'**. Potete anche aggiungere un po' più di flessibilità usando **WHERE** e **LIKE** come in **WHERE lname LIKE 'v%'**; questo ritornerà gli utenti il cui cognome comincia con una V. Notate che le stringhe da cercare, per default, NON sono case-sensitive.

```
SELECT * FROM users WHERE uid = 1;
SELECT * FROM users WHERE lname = 'Rossi';
SELECT * FROM users WHERE lname LIKE 'v%';
```

Ora avete una conoscenza sufficiente per essere pericolosi e avventurarvi nel PHP per il vostro sito Web. Ricorderete di avere un link a una pagina di ricerca che finora non esiste, ma con quello che sapete ora potete prepararla. Per prima cosa, non dimenticatevi di chiudere il prompt dei comandi di MySQL scrivendo **exit**.

Creare la pagina di ricerca HTML

Lavorando nella directory del filesystem **/var/www** che avete configurato come Document Root del server Nginx,

Tip

I file PHP come **config.php** sono inclusi da **include_path**, che per default include la directory corrente; percorsi aggiuntivi possono essere aggiunti usando la funzione **PHP set_include_path()**. Se gestite il server potete anche impostarli in modo permanente nel **php.ini** con la direttiva **include_path**.



› L'output di ss può mostrare le porte in uso su Raspberry Pi. È simile all'output di netstat ma ss è dedicato a questo

create la pagina **/var/www/search.html**. Sarà una pagina in puro XHTML che in seguito sfrutterà una pagina PHP ad hoc. La pagina Web che creerete avrà un semplice form che vi permetterà di cercare utenti in base al cognome. La pagina dovrebbe risultare simile al seguente codice:

```
<?xml version=”1.0” encoding=”UTF-8”?> <!DOCTYPE
 html PUBLIC “-//W3C//DTD XHTML 1.0 Strict//EN”
  “http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd”>
<html xmlns=”http://www.w3.org/1999/xhtml”>
 <head>
   <title>Ricerca</title>
   <meta name=”Author” content=“LinuxPro” />
   <meta name=”Description” content=“Ricerca
   Database” />
   <link rel=”stylesheet” href=”style.css” />
 </head>
 <body>
   <div id=”pagina”>
     <div id=”header” class=”frame”>
       <h1>Ricerca Database</h1>
     </div>
     <div id=”menu” class=”frame”>
       <ul>
         <li><a id=”home” href= “/index.html”>Home
       </a></li>
         <li><a id=”search” href= “/search.html”>Ricerca
       </a></li>
       </ul>
     </div>
     <div id=“contenuto” class=”frame”>
       <p>Ricerca in MySQL con PHP</p>
       <form method=”get” action=”/search.php”>
         <label for=”last”>Inserisci un cognome:</label>
         <input type=”text” name=”last”></input><br />
         <input type=”submit” value=“Ricerca”
name=”submit”> </input>
       </form>
     </div>
   </div>
 </body>
</html>
```

Potete vedere che il form usa il metodo **GET** e l'azione è riferita a search.php nella Document Root del server: è quest'ultima pagina che si connette al server MySQL, esegue la ricerca SQL e mostra il risultato. A questo punto dovreste poter accedere al vostro sito Web e vedere la pagina di ricerca, anche se il submit non funzionerà.

› Potete utilizzare il form HTML per cercare nel database, una volta implementato il codice PHP per il backend

Codice di ricerca in PHP

Per ultimare il progetto dovete creare la pagina PHP che eseguirà e mostrerà il risultato della ricerca. Create dunque **/var/www/search.php** in modo simile al codice di seguito:

```
<!DOCTYPE html PUBLIC “-//W3C//DTD XHTML 1.0 Strict//EN”
        “http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd”>
<html xmlns=”http://www.w3.org/1999/xhtml”>
<head>
        <title>Risultato</title>
        <meta name=”Author” content=”LinuxPro” />
        <meta name=”Description” content=”Risultato
Database” />
        <link rel=”stylesheet” href=”style.css” />
</head>
<body>
        <div id=”pagina”>
                <div id=”header” class=”frame”>
                        <h1>Risultati ricerca</h1>
                </div>
                <div id=”menu” class=”frame”>
                        <ul>
                                <li><a id=”home” href= “/index.
html”>Home</a></li>
                                <li><a id=”search” href= “/search.
html”>Ricerca</a></li>
                        </ul>
                </div>
                <div id=”contenuto” class=”frame”>
                        <p>Risultati:</p>
                        <table>
                                <?php
                                $host = ‘localhost’;
                                $user = ‘root’;
                                $pwd = ‘Password1’;
                                $db = ‘contact’;
                                $dbh = mysqli_
connect($host,$user,$pwd,$db) or die( “ooooops” );
                                $last = $_GET[‘last’];
                                if ( $last != ‘’ ) {
                                        $query=”SELECT * from users
WHERE lname LIKE ‘${last}%’”;
                                } else {
                                        $query=”SELECT * from users”;
                                }
                                $result=mysqli_query($dbh,$query);
                                while ($row = mysqli_fetch_
array($result,MYSQLI_ASSOC)){
                                        $uid = $row[‘uid’];
                                        $fn = $row[‘fname’];
                                        $sn = $row[‘lname’];
                                        print “<tr><td>” . $uid . “</
td><td>” . $fn . “</td><td>” . $sn . “</td></tr>”;
                                }
                                mysqli_free_result($result);
                                mysqli_close($dbh);
                                ?>
                        </table>
                </div>
        </div>
</body>
</html>
```

La sezione principale del codice su cui focalizzarvi è nel div



› **Anche se il database non è accessibile dal mondo esterno la pagina Web può fungere da portale per i vostri dati**

"contenuto". All'interno di questo div viene visualizzato il paragrafo con "Risultati:" e viene aperto un elemento table, seguito dal codice PHP. Una volta che l'elemento PHP è chiuso, chiudete l'elemento table. Il codice PHP genererà la tabella per voi. All'interno della tag PHP, le prime cinque righe stabiliscono una connessione al database. In realtà sarebbe meglio servire tali linee da un file separato con un'istruzione include. In questo modo il nome utente e la password non sarebbero nella vostra pagina, chiaro buco di sicurezza. Per semplicità l'esempio le integra, ma nel vostro codice dovrebbero essere: **include_once('config.php');**. Il codice sostituito con questa riga avrà i propri tag php all'interno del file **config.php**. La variabile ritornata dalla connessione sarà "**$dbh**" e nel resto del codice rappresenterà la connessione al database. Estraete quindi dalla URI il valore dell'ultimo elemento nell'array **$_GET** e lo salvate come variabile locale, battezzata **$last**. Usando il blocco condizionale **if** controllate se **$last** ha o meno un valore. Senza un valore, la query cercherà tutti gli utenti; con un valore, cercherà tutti gli utenti il cui cognome inizia per la stringa fornita nella form. La query viene eseguita con l'uso della funzione **mysqli_query** e i risultati registrati nella variabile chiamata originalmente **$result**. Per popolare la tabella si sfrutta il ciclo **while** iterando sulla variabile **$result** e aggiungendo righe per ogni record trovato. I tag HTML che delimitano la cella sono integrate nel codice PHP. L'ultimo elemento della riga è il cognome dell'utente e dunque chiudete la riga stessa con **</tr>**. In questo modo potete aggiungere dinamicamente alla tabella definita in HTML tante righe quante ne richiede il risultato della query. Una volta usciti dal ciclo while rilasciate la memoria utilizzata dal set di risultati di MySQL e chiudete la connessione al database. Il controllo delle risorse è di vitale importanza per l'affidabilità della soluzione e dovete assicurarvi che entrambe le operazioni vengano eseguite. Nota: questo è un modo semplice per implementare questa operazione ed è pensato per farvi familiarizzare su come PHP può accedere al database MySQL. Pensatelo come punto di partenza e bozza. Un sistema di produzione richiederà più feedback e sicurezza per essere efficace.

Testare la soluzione

Con tutti i mattoni al loro posto potete ora controllare che sia **search.html** sia **search.php** siano nella directory **/var/www** del server Nginx. Aggiornate il browser Web e navigate alla pagina di ricerca. Se tutto è a posto dovreste vedere il vostro magnifico sito. Provandolo scoprite che potete cercare utenti con o senza specificare il cognome. Inserendo la lettera V nel campo cognome avrete di ritorno i due utenti il cui cognome inizia per V. Se funziona datevi una pacca sulla spalla: avete creato qualcosa di magnifico e si spera sia solo il primo passo per un mondo di pagine Web dinamiche. Se non dovesse funzionare, controllate i file di log nel server, che trovate in **/var/log/nginx/error.log**. LXP

Vincoli per chiave primaria, unica ed esterna

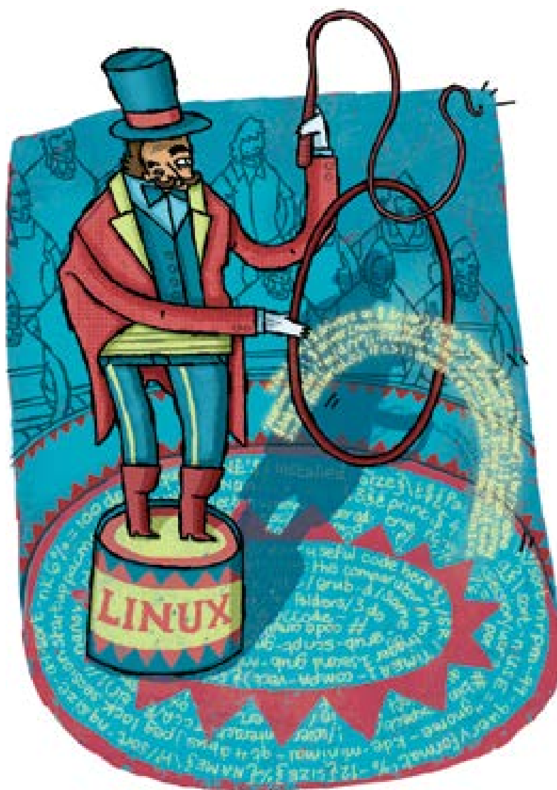
In questo esempio avete visto l'uso di **PRIMARY KEY**. Per ogni tabella potete avere una sola chiave primaria; l'uso di una chiave non è indispensabile ma decisamente consigliato per identificare univocamente ogni riga nella tabella. Una chiave primaria può essere definita su una o più colonne in una tabella, l'idea è quella di creare un valore unico per la chiave. Solitamente può essere costituita da una sola colonna ma a volte deve contenerne più di una. Per esempio, la tabella di una biblioteca può avere una chiave primaria definita sulle colonne **ISBN** e **Numero Copia**. Di suo l'ISBN non è unico e nemmeno la copia, ma assieme producono un valore unico che identifica la copia di un libro. Analogamente alla chiave primaria esiste un vincolo UNIQUE: diversamente dalla chiave primaria possono essercene più di uno in una singola tabella, ma ogni valore dev'essere, come per la chiave primaria, unico. Un altro tipo di vincolo di chiave non utilizzato nell'esempio è FOREIGN KEY. Questo assicura che il dato inserito in una colonna esista anche in un'altra. Per esempio in una tabella clienti, ha senso rendere la colonna Città che costituisce l'indirizzo dipendente da un elenco fisso di città. Implementerete quindi il vincolo tra la tabella clienti e la tabella città, assicurandovi che non possano essere inseriti valori nella prima che non esistano nella seconda.

Wireshark Impostate l'analizzatore di Rete per catturare e filtrare il vostro traffico

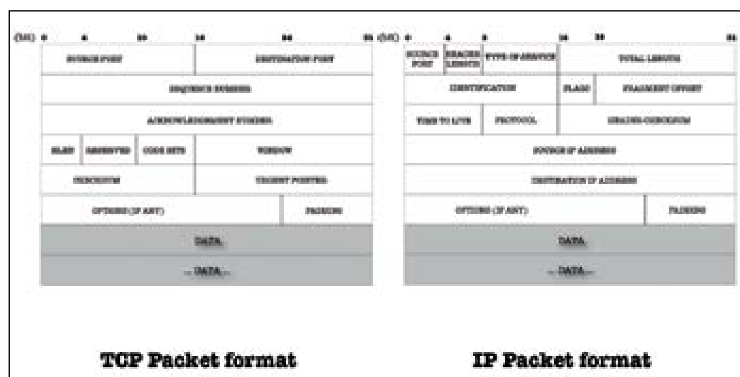
Analizzare il traffico

Ecco tutto quello che dovete sapere per cominciare a usare Wireshark e analizzare tre tipi di traffico di rete





Wireshark è un analizzatore di protocolli di rete molto conosciuto e potente, sviluppato da Gerald Combs. Wireshark è nato nel giugno 2006 quando Combs ha rinominato il tool **Ethereal** (creato sempre da lui) perché stava cambiando lavoro e non poteva più utilizzare il nome originale. Oggi la maggior parte delle persone utilizza Wireshark ed Ethereal è considerato storia. La vostra distribuzione Linux ha anche un pacchetto pronto per l'installazione, quindi andate e installate!



› **Il formato dei pacchetti TCP e IP**

Potreste chiedervi cosa rende Wireshark diverso da altri analizzatori di rete, a parte la gratuità, e perché non utilizzare semplicemente **tcpdump** per la cattura dei pacchetti. Il vantaggio principale di Wireshark è la sua interfaccia grafica che semplifica di molto la complessità dei dati di rete. I principianti prima di capire come funziona Wireshark dovranno capire come funziona il traffico di rete. Lo scopo di questo articolo è quindi di fornire una introduzione a TCP/IP per permettervi di trarre utili conclusioni sul traffico di rete che andrete ad analizzare. Se lanciate Wireshark come utente normale non potrete usare alcuna delle interfacce di rete per la cattura a causa dei permessi standard di Unix che hanno. È decisamente più utile lanciare Wireshark come root (**sudo wireshark**) quando dovete catturare dati e come utente normale quando li analizzate. Alternativamente potete catturare traffico di rete usando **tcpdump** da riga di comando come root e analizzarlo in seguito con Wireshark. Tenete a mente che in una rete piuttosto carica la cattura con Wireshark può rallentare una macchina o, peggio, potreste non riuscire a catturare tutto perché Wireshark richiede più risorse di sistema di un programma a riga di comando. In tali casi l'utilizzo di tcpdump per la cattura è la soluzione più saggia.

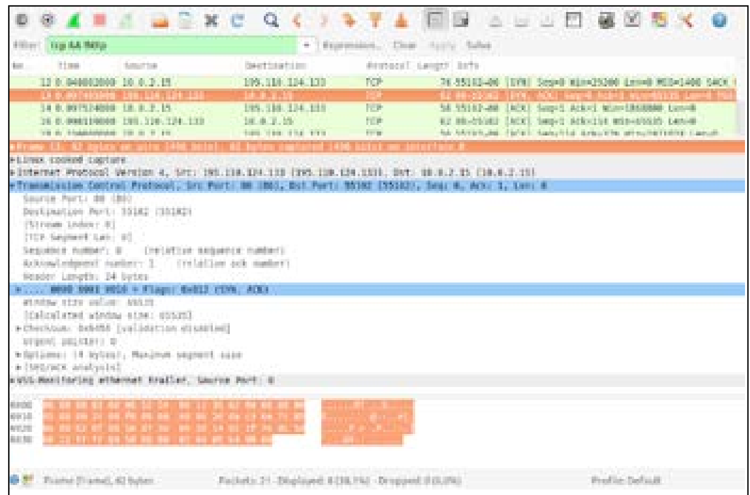
Catturare dati di rete

Il modo più semplice di iniziare a catturare pacchetti di rete è selezionare la vostra interfaccia preferita dopo il lancio di Wireshark e premere **Start**. Wireshark mostrerà i dati di rete sul vostro schermo a seconda del traffico. Notate che potete selezionare più di un'interfaccia. Se non conoscete nulla riguardo TCP, IP o gli altri protocolli TCP/IP, potreste trovare l'output complicato e difficile da leggere o capire. Per fermare il processo di cattura scegliete **Capture** ➟ **Stop** dal menu. Alternativamente potete premere la quarta icona da sinistra, quella con il quadrato rosso (scorciatoia per "Ferma la cattura in corso") sulla barra icone principale (la posizione esatta dipende dalla vostra versione di Wireshark). Questo pulsante può essere premuto solo quando state catturando. Utilizzando il metodo descritto per la cattura non potete modificare alcuna delle opzioni di default di Wireshark. Potete farlo invece selezionando **Capture** ➟ **Options** dal menu. Qui potete selezionare l'interfaccia di rete, vedere il vostro indirizzo IP, applicare filtri di cattura, mettere la scheda in modalità promiscua e salvare i dati catturati in uno o più file. Potete anche scegliere di fermare la cattura dopo un dato numero di pacchetti o un tempo prefissato, o ancora una dimensione definita di dati (in byte). Wireshark per default non salva i dati catturati ma potete salvarli in seguito. È buona norma prima salvare e poi esaminare i pacchetti di rete, a meno che non ci sia una buona ragione per non farlo. Wireshark vi permette di leggere e analizzare dati di rete già catturati da un vasto campione di formati, inclusi **tcpdump**, **libpcap**, **snoop** di Sun, **nettl** di HP, file di testo K12, ecc. Questo significa che potete leggere praticamente qualsiasi formato di dati catturati; analogamente Wireshark vi permette di salvare i vostri dati in una varietà di formati. Potete anche utilizzare Wireshark

per convertire un file da un dato formato a un altro. Potete anche esportare un file esistente come testo semplice dal menu **File**. Questa opzione è principalmente per la modifica manuale o per utilizzarlo come input per un altro programma. C'è un'opzione che vi permette di stampare i vostri pacchetti, che sebbene non abbia un evidente utilizzo nella vita reale può essere utile per scopi didattici.

Filtri di visualizzazione

Mentre i filtri di cattura vengono applicati durante la cattura dei dati di rete e permettono a Wireshark di scartare traffico di rete al quale non siete interessati, i filtri di visualizzazione sono applicati dopo la cattura e 'nascondono' il traffico senza eliminarlo.
Potete sempre disabilitare un filtro di visualizzazione e tornare a vedere i vostri dati. In generale, i filtri di visualizzazione sono considerati più utili e versatili dei filtri di cattura perché non è detto che sappiate in anticipo quello che catturerete o che vorrete esaminare. Applicare filtri al momento della cattura tuttavia può farvi risparmiare tempo e spazio su disco e queste sono le ragioni principali per cui usarli. Wireshark vi evidenzierà quando un filtro di visualizzazione è sintatticamente corretto con uno sfondo verde chiaro. Quando la sintassi è sbagliata, lo sfondo diventa rosa.
I filtri di visualizzazione supportano operatori logici e di confronto. I filtri **http.response.code == 404 && ip.addr == 192.168.1.1** mostrano il traffico che arriva verso o dall'indirizzo 192.168.1.1 e contemporaneamente ha il codice risposta HTTP 404 (Not Found). I filtri **!bootp && !ip && !arp** escludono il traffico BOOTP, IP e ARP dall'output. I filtri di visualizzazione **eth.addr == 01:23:45:67:89:ab && tcp.port == 25** mostrano il traffico verso e dal dispositivo con l'indirizzo MAC 01:23:45:67:89:ab che usa la porta TCP 25 nelle connessioni in entrata e in uscita. Tenete a mente che i filtri di visualizzazione non risolvono magicamente i problemi. Sono strumenti estremamente utili se usati correttamente ma dovete comunque interpretare i risultati, trovare il problema e pensare da voi alle possibili soluzioni.
Nel definire le regole ricordate che l'espressione **(ip.addr != 192.168.1.5)** non significa che nulla dei campi ip.addr possa contenere l'indirizzo IP 192.168.1.5, bensì che i campi ip.addr non *dovrebbero* contenere l'indirizzo IP 192.168.1.5. Quindi, l'altro campo ip.addr può essere uguale a 192.168.1.5. Potete pensarlo come "esiste un campo ip.addr che non sia 192.168.1.5".
Il modo corretto di filtrare è **!(ip.addr == 192.168.1.5)**.
È un errore abbastanza comune. Ricordate anche che gli indirizzi MAC sono veramente utili nel tracciare una macchina precisa sulla vostra LAN dal momento che l'IP di una macchina può cambiare se usa DHCP ma il suo indirizzo MAC è quasi sempre statico.
Date un'occhiata al sito di riferimento per i filtri legati al traffico TCP su **http://bit.ly/WireSharkTCP**. Per la lista di tutti i nomi di capi disponibili riguardo il traffico UDP visitate il sito **http://bit.ly/WireSharkUDP**.



› **I tre pacchetti (SYN, SYN+ACK e ACK) di un handshake TCP**

Riguardo TCP/IP, TCP e IP

TCP/IP è il protocollo più usato per connettere computer ed è talmente legato a Internet che è molto difficile parlare di TCP/IP senza parlare di Internet e viceversa. Ogni dispositivo che ne fa uso ha:

» **Un indirizzo IP** Questo indirizzo dev'essere unico almeno nella rete locale.
» **Una maschera di rete** Utilizzata per dividere grosse reti IP in reti più piccole in relazione alla rete attuale.
» **Uno o più server DNS** Usati per tradurre un indirizzo IP in un formato leggibile da un umano e viceversa.
» **Un default gateway** Opzionale, usato per comunicare con dispositivi al di fuori della rete locale. Un default gateway è un dispositivo di rete a cui viene inviato un pacchetto TCP/IP quando il mittente non sa a chi altro consegnarlo. Ogni servizio TCP ascolta su una porta unica a ogni macchina. Una macchina che supporta il protocollo HTTP, il protocollo che serve WWW, è chiamata anche server HTTP. Allo stesso modo esistono server FTP, DNS, ecc.

La connessione tra due macchine che usano TCP/IP è identificata dalle due rispettive coppie indirizzo IP / porta sorgente e destinazione. Un pacchetto TCP (vedete l'immagine in apertura) può essere usato per stabilire connessioni, trasferire dati, inviare riscontri (acknowledgement), indicare il buffer che contiene i dati in ingresso (Window Size) e chiudere connessioni. Ogni segmento TCP ha una parte di header e una parte di dati.

L'handshake a 3 vie di TCP

TCP fornisce un servizio di flusso di byte affidabile orientato alla connessione. È un protocollo full-duplex, ovvero ogni connessione TCP supporta una coppia di flussi di byte, uno in ogni direzione.

Tip

Il fatto che il protocollo FTP utilizzi solitamente la porta 21 non significa che non possa utilizzarne una diversa. In altre parole, non affidatevi ciecamente alle porte conosciute del traffico TCP/IP.

Il protocollo TCP

TCP sta per *Transmission Control Protocol*. La caratteristica principale di TCP è che è affidabile e si assicura che un pacchetto sia consegnato. Se non c'è prova di consegna di un pacchetto, lo reinvia. Il software TCP trasmette dati tra le macchine utilizzando segmenti (chiamati anche pacchetti TCP). TCP assegna un numero di sequenza ad ogni byte trasmesso e si aspetta un riscontro positivo (ACK) dallo stack TCP ricevente. Se non si riceve l'ACK entro un intervallo di timeout, il dato viene ritrasmesso e il pacchetto originale considerato non consegnato. Lo stack TCP ricevente utilizza i numeri di sequenza per riordinare i segmenti quando arrivano fuori ordine e per eliminare segmenti duplicati. L'header TCP include i campi **Porta Sorgente** e **Porta Destinazione**. Questi due campi, assieme agli indirizzi del mittente e del destinatario, sono combinati per identificare univocamente ogni connessione TCP. Le porte aiutano gli stack TCP/IP nei dispositivi connessi (PC e router) a distribuire il traffico tra più programmi in esecuzione su un singolo dispositivo. Se un servizio vuole essere visto come affidabile solitamente è basato su TCP, altrimenti su IP. Come potete immaginare, tuttavia, l'affidabilità ha un suo costo e non è sempre desiderabile.

```
384 1.188311000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
385 1.191135000 64:70:02:ad:e9:44 b8:e8:56:34:a1:c8  ARP   60 10.67.93.1 is
386 1.193465000 d0:27:88:1d:d6:fb b8:e8:56:34:a1:c8  ARP   60 10.67.93.21 is
387 1.194804000 d0:27:88:1d:15:20 b8:e8:56:34:a1:c8  ARP   60 10.67.93.20 is
388 1.196202000 00:1a:92:44:d7:67 b8:e8:56:34:a1:c8  ARP   60 10.67.93.22 is
389 1.210704000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
390 1.210706000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
391 1.210707000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
392 1.210707000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
393 1.210708000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
394 1.210708000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
395 1.210709000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
396 1.222069000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
397 1.222070000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
398 1.222071000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.
399 1.222072000 b8:e8:56:34:a1:c8 Broadcast          ARP   42 Who has 10.67.

    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: d0:27:88:1d:d6:fb (d0:27:88:1d:d6:fb)
    Sender IP address: 10.67.93.21 (10.67.93.21)
    Target MAC address: b8:e8:56:34:a1:c8 (b8:e8:56:34:a1:c8)
    Target IP address: 10.67.93.11 (10.67.93.11)

0000  b8 e8 56 34 a1 c8 d0 27  88 1d d6 fb 08 06 00 01   ..V4...' ........
0010  08 00 06 04 00 02 d0 27  88 1d d6 fb 0a 43 5d 15   .......' .....C].
0020  b8 e8 56 34 a1 c8 0a 43  5d 0b 00 00 00 00 00 00   ..V4...C ].......
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

Ready to load or capture                          Packets: 531 · Displayed: 531 · M
```

› **Parte di uno scan ping Nmap su una LAN come catturato da Wireshark**

Il termine "orientato alla connessione" indica che le due applicazioni utilizzanti TCP devono prima stabilire una connessione TCP tra di loro prima dell'effettivo scambio di dati. L'header TCP include un campo flag a 6 bit utilizzato per le informazioni di controllo tra i due capi della comunicazione. Le flag possibili includono SYN, FIN, RESET, PUSH, URG e ACK. Le flag SYN e ACK sono usate per l'handshake iniziale a tre vie come vedrete tra un momento. La flag RESET indica la volontà del ricevente di abortire la connessione. L'handshake TCP a tre vie funziona più o meno in questo modo: il client invia un pacchetto TCP SYN al server e l'header TCP include un numero di sequenza arbitrario nel pacchetto stesso. Il server risponde con un pacchetto TCP (SYN, ACK) che include il numero di sequenza della direzione opposta e un riscontro del numero di sequenza precedente. Infine, per stabilire definitivamente la connessione TCP, il client risponde con un pacchetto TCP ACK per riscontro del numero di sequenza inviato dal server. Dopo l'handshake, la connessione è stabilita e pronta all'invio e alla ricezione di dati. Il traffico per questo esempio è stato prodotto lanciando il comando seguente:

```
$ wget http://www.linuxpro.it/
```

Dopo alcune richieste DNS, ARP e ICMP, comincia l'handshake TCP a tre vie. L'indirizzo del client è 10.0.2.15 e l'IP destinazione è 195.110.124.133. Un semplice filtro di visualizzazione (**tcp && !http**) riduce la visualizzazione dei 21 pacchetti catturati a 8. I numeri utilizzati per le sequenze sono consecutivi perché al momento l'host non stava facendo alcun altra attività di rete, ma raramente è così.

Tip

Quando impostate la vostra scheda di rete in modalità promiscua, le permettete di accettare e leggere ogni pacchetto di rete che arriva anche se la destinazione è un altro dispositivo sulla rete. I pacchetti di rete giungono comunque al destinatario.

Ping scan

Questa parte esaminerà il traffico di rete prodotto da **Nmap** quando effettua un ping scan. I ping scan in LAN sono eseguiti utilizzando il protocollo ARP. Gli host al di fuori della LAN sono interamente scansionati con il protocollo ICMP, quindi se fate uno scan con **Nmap** al di fuori della LAN, il traffico sarà diverso da quello presentato. Nell'esempio sottostante, il comando scansiona 255 indirizzi IP, da 192.168.1.1a 192.168.1.255. I risultati mostrano che al momento dell'esecuzione solo 7 macchine erano attive o, più precisamente, solo 7 macchine hanno risposto allo scan Nmap:

```
$ sudo nmap -sP 192.168.1.1-255

Starting Nmap 6.46 ( http://nmap.org ) at 2014-12-24 13:01
CET
Nmap scan report for 192.168.1.100
Host is up (0.00076s latency).
MAC Address: xx:yy:zz:D6:93:kk (Raspberry Pi Foundation)
Nmap scan report for 192.168.1.129
Host is up (0.010s latency).
MAC Address: xx:yy:zz:85:00:kk (Apple)
Nmap scan report for 192.168.1.130
Host is up (0.010s latency).
MAC Address: xx:yy:zz:A8:B5:kk (Unknown)
Nmap scan report for 192.168.1.131
Host is up (-0.10s latency).
MAC Address: xx:yy:zz:AE:F3:kk (LG Electronics)
Nmap scan report for 192.168.1.133
```

```
Host is up (-0.10s latency).
MAC Address: xx:yy:zz:00:33:kk (Syncmold Enterprise)
Nmap scan report for 192.168.1.254
Host is up (-0.099s latency).
MAC Address: xx:yy:zz:42:5D:kk (Pirelli Tyre S.p.A.)
Nmap scan report for 192.168.1.128
Host is up.
Nmap done: 255 IP addresses (7 hosts up) scanned in 3.31 seconds
```

Lo scopo di questo test è semplicemente di capire se un IP è attivo o meno. Quello che è importante per Nmap in un ping scan non sono i dati effettivamente contenuti nei pacchetti ricevuti ma, più semplicemente, l'esistenza di un pacchetto di risposta. Dal momento che tutto il traffico è in LAN ogni dispositivo usa il proprio indirizzo MAC nella risposta, quindi vedete solo quelli sia nel campo sorgente sia destinazione. La presenza di una risposta indica a Nmap che la macchina è attiva e funzionante. Dal momento che l'indirizzo MAC contiene informazioni riguardo il produttore del dispositivo, Nmap ve le mostra. Nmap calcola anche il ritardo medio (o latenza), che vi offre una stima accurata del tempo necessario perché il primo pacchetto (inviato da Nmap) arrivi a destinazione più il tempo necessario al pacchetto di risposta per tornare a Nmap. Una latenza elevata non è cosa buona e deve essere certamente analizzata.

Analizzare il traffico DNS

Le query DNS sono piuttosto frequenti nelle reti TCP/IP. Una query DNS crea poco traffico ed è quindi un ottimo esempio per scopi didattici. Potete usare il comando seguente per generare il traffico DNS necessario da esaminare:

```
$ host -t ns linuxpro.it
linuxpro.it name server ns2.register.it.
linuxpro.it name server ns1.register.it.
```

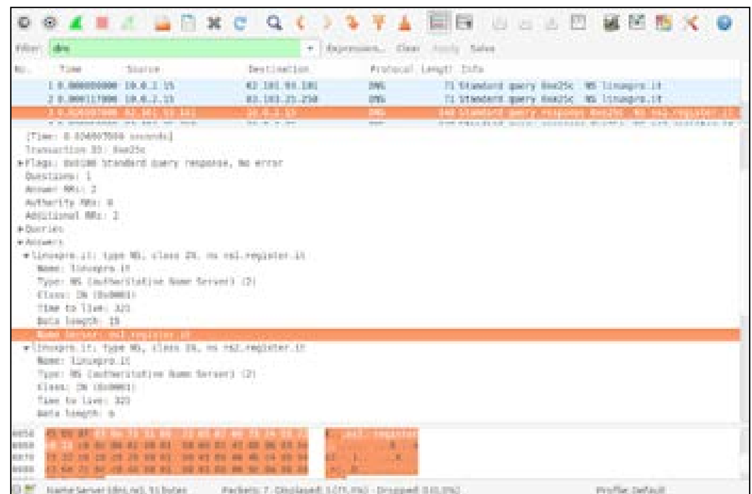
Sono necessari in totale due pacchetti: uno per l'invio e uno per la risposta della query DNS.
Il primo pacchetto è il numero 1 e il secondo il numero 3.
È stato usato un filtro di visualizzazione (DNS) per minimizzare i dati visualizzati e ottenere le informazioni significative.
Il protocollo usato è UDP (*User Datagram Protocol*) e le informazioni sono state consegnate senza errori, come indicato nella sezione Flags. Potete anche notare dal timestamp della query (0.000000000) e della risposta (0.026907000) che il servizio DNS ha risposto in tempo breve ed è dunque affidabile. Il server DNS contattato ha l'indirizzo 62.101.93.101, come potete vedere dall'indirizzo IP destinazione del primo pacchetto. Lo stesso server DNS ha risposto alla query, come potete vedere dall'indirizzo IP sorgente del secondo pacchetto. La riga 'Answer RRs: 2' vi informa che ci sono state due risposte per la query.
Con il tempo tali informazioni diventeranno normali per voi. UDP usa il protocollo IP sottostante per trasportare un messaggio da una macchina a un'altra e offre la stessa consegna non affidabile e senza connessione di IP. Non usa i riscontri per assicurarsi che i messaggi arrivino, non ordina i messaggi in entrata e non offre feedback per controllare la frequenza con la quale le informazioni scorrono tra le macchine, per cui i messaggi UDP possono essere persi, duplicati, o arrivare fuori ordine. Per di più, i pacchetti possono arrivare più velocemente di quanto il ricevente possa processarli. La porta di destinazione del primo pacchetto è la 53, ovvero la porta conosciuta per il servizio DNS.
La parte UDP del secondo pacchetto mostra la porta usata per la risposta:

```
User Datagram Protocol, Src Port: 53 (53), Dst Port: 31300 (31300)
Source Port: 53 (53)
Destination Port: 31300 (31300)
Length: 114
Checksum: 0x4ea1 [validation disabled] [Stream index: 0]
```

Come succede per la maggior parte degli strumenti, più usate Wireshark più aumenterà la vostra efficienza, quindi continuate a far pratica! LXP

Tip

C'è anche una versione console di Wireshark chiamata **tshark**. I due vantaggi principali di tshark sono che può essere integrato in script e utilizzato su connessione SSH. Lo svantaggio più grosso è che non ha una GUI. Tshark può anche sostituire completamente **tcpdump**.



› Ecco come Wireshark mostra il traffico di una query DNS dopo l'applicazione di un filtro Display. Notate il colore verde attorno a DNS che ne mostra la validità

Il protocollo IP

IP sta per *Internet Protocol*. La caratteristica principale di IP è che non è un protocollo affidabile per design. Non affidabile significa che i pacchetti potrebbero non raggiungere la destinazione per varie ragioni, inclusi errori di trasmissione, errori hardware e congestione di rete. Le reti possono anche consegnare i pacchetti non in ordine, dopo un ritardo sensibile o duplicati. Un programmatore può tuttavia disegnare applicazioni affidabili che utilizzino IP implementando il proprio codice per il controllo d'errore ma non è un compito triviale. Quando l'informazione non richiede molti pacchetti, utilizzare un protocollo basato su IP è più efficiente di TCP, anche se dovete ri-trasmettere un pacchetto di rete, dal momento che viene risparmiato l'overhead dell'handshake. IP incapsula i dati che viaggiano in una rete TCP/IP, dal momento che è responsabile dalla consegna dei pacchetti dal sorgente alla destinazione secondo gli indirizzi IP. Tali indirizzi devono essere determinati per far giungere i pacchetti a destinazione; questo compito è svolto principalmente da dispositivi riconosciuti come router, ma ogni dispositivo TCP/IP deve poter fare un routing basilare. Ogni indirizzo IP è una sequenza di quattro numeri di 8 bit separati da punti. Ogni numero ha un valore tra 0 (=2^0-1) e 255 (=2^8-1). Esempi di indirizzi IP sono 10.25.128.254, 213.201.43.23 e 192.168.1.200. IPv6 è stato sviluppato da IETF con lo scopo di risolvere il problema dell'esaurimento degli indirizzi IPv4. IP usa indirizzi a 32 bit mentre IPv6 a 128 bit, offrendo oltre 7.9x1028 più indirizzi di IPv4.

Da Python a codice nativo

Cython è un compilatore C per Python: eccovi una veloce guida per rendere il vostro codice fino a 12 volte più veloce!



Potete aver già sentito parlare di **Cython**. Solitamente salta fuori nelle discussioni sulla velocità di Python. Cython è un linguaggio di programmazione che estende Python con alcune keyword e costrutti così da potersi avvantaggiare dei tipi C nativi. Ciò significa che se dovete calcolare '1+2', Cython può riconoscere i due interi e compilare in C:

```
int result = 1 + 2;
```

Questo ha alcuni interessanti effetti secondari che vedrete più avanti nel tutorial. Per il momento pensate a Cython come a un Python in grado di utilizzare tipi reali e codice Python. Per prima cosa, per installare Cython potete usare gestori di pacchetti come **apt-get** oppure utilizzare **pip** per maggiore portabilità e aggiornamento:

```
$ pip install cython
```

Vi serviranno anche gli header di sviluppo Python e le librerie così da ottenere lo script **python-config**.
Su Ubuntu potete ottenerle con

```
sudo apt-get install python-dev
```

oppure su Fedora

```
sudo yum install python-devel
```

Ora siete pronti a usare Cython. Per vederlo subito in azione date un'occhiata alla classica funzione fibonacci in Python:

```
def fib(n):
	if n == 0:
		return 0
	elif n == 1:
		return 1
	else:
		return fib(n-1) + fib(n-2)
```

Questo è il codice Python con il quale di sicuro sarete già familiari. Ritorna l'ennesimo elemento della sequenza di Fibonacci. Ora in Cython potete riscrivere semplicemente la dichiarazione di funzione così:

```
cdef fib(int n):
	if n == 0:
		return 0
	elif n == 1:
		return 1
	else:
		return fib(n-1) + fib(n-2)
```

Potete notare che risultano praticamente identiche, tranne la dichiarazione di funzione cython (cdef) e la dichiarazione del tipo del parametro (int). Cython utilizza queste informazioni per generare codice ottimizzato, dal momento che sapete di certo che state trattando un intero e che lo scoping dei dati all'interno vi permette di usare gli stack C. Per compilare ed eseguire questo codice, tuttavia, il processo è un filo più complesso. Invece di lanciare semplicemente l'interprete Python sul codice, dovete lanciare il compilatore Cython e usare quindi GCC per compilare il codice in un eseguibile:

```
$ cython --embed fib2.pyx -o fib.c
$ gcc -g -O2 -o fib2 fib2.c `python-config --includes --ldflags``
```

Ora potete confrontare i tempi di esecuzione delle due implementazioni della funzione per i primi 30 numeri, per farlo vi basta aggiungere un pizzico di codice Python per chiamare la funzione a ogni versione, dato che Cython compilerà perfettamente!

```
result = []
for i in range (30):
	result.append(fib(i))
print result
```

Confrontate quindi quanto ci mette a lanciare ogni versione:

```
$ time ./fib1.py # versione python puro
<output>
real 0m0.537s
user 0m0.529s
sys 0m0.006s
$ time ./fib2 # versione cython
<output>
real 0m0.045s
user 0m0.040s
sys 0m0.005s
```

La versione Cython ha un notevole incremento, più o meno 12 volte più veloce! Come detto, l'utilizzo dei tipi nativi implica che il compilatore Cython sa come bypassare interamente il runtime python per la valutazione del codice e genera codice C per farlo. L'interessante effetto collaterale è dato dal fatto

Tip

Cython non supporta gli import tradizionali, dal momento che un singolo file cython è in sé un intero modulo Python. Quello che potete fare è un import in stile C/C++ nel file Cython principale, per esempio: **include morecode.pxi**.

Ogni cosa è un modulo

In Cython il singolo file sorgente rappresenta un intero modulo Python. Nello scrivere grosse estensioni Cython si usa la direttiva **include** per rendere il codice più modulare. Nel disegnare il vostro modulo di estensione Cython dovete pensare a come questo verrà eseguito: come vedete dagli esempi viene passato l'argomento di integrazione al compilatore Cython. Quello che succede è la generazione di un metodo **main** per il programma principale, cosicché l'esecuzione inizi come vi aspettereste nel vostro codice Cython. Non siete tenuti a farlo, tuttavia, e potete renderlo come un normale modulo Python (lo vedete quando non passate l'argomento e lo compilate come libreria condivisa). Per interagire con questa libreria, comunque, dovete stare attenti alla nomenclatura e ai nomi. Se il file principale Cython è MioModulo.pyx dovrete compilarlo in MioModulo.c e infine in MioModulo.so via gcc. Nella shell Python potrete poi importarlo semplicemente con un **import MioModulo** fintantoché il modulo sta nel vostro PYTHONPATH. Potete anche compilarlo come libreria condivisa o semplicemente collegarlo a un programma C/C++, ma dovrete inizializzare il runtime di Python e il modulo stesso prima di interagire con il codice Cython, pena un crash. Potete vederlo nell'esempio di flask.

che, producendo codice C, potete richiamarlo direttamente all'interno di codice e strutture dati in C/C++ puro, e addirittura richiamarlo in seguito ancora dentro il codice Python! Questo potenzialmente risolve un problema nella scrittura di sistemi complessi in C/C++ rispetto a JVM, dove in rapporto manca molta astrazione dinamica: in questo modo potete dividere il software in C/C++ con Python per gestire comportamenti dinamici nel controllo delle strutture dati. Anche per lo unit testing del codice C/C++ Cython offre un livello comodo sul quale astrarre o impostare l'ambiente di test. Per vederlo in azione integrerete una webapp Flask in un'applicazione C e vi servirete alcune strutture dati in memoria in formato JSON. Questa tecnica è utilizzata anche in sistemi economici a elevata performance con una dashboard Web per visualizzare tabelle in memoria condivisa tramite delle api C++ per l'accesso ai dati, invece di codice Python. Ci sono diverse parti per farlo, tuttavia per prima cosa compilerete l'app flask helloworld dall'homepage su **http://flask.pocoo.org**, chiaramente dopo averlo installato:

```
$ pip install flask
```

Quindi Hello World:

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
        return "Ciao mondo!"

if __name__ == "__main__":
        app.run()
```

Come potete vedere Cython può compilare codice Python puro in C ma utilizzerà il runtime Python. Per compilarlo userete ancora il compilatore Cython per generare il rispettivo codice C su cui lanciare gcc per ottenere l'eseguibile:

```
$ cython -2 --embed app.pyx -o app.c
$ gcc -g -O2 -o app app.c `python.config --includes
--ldflags`
```

Giusto per chiarezza, le opzioni utilizzate nell'invocare il compilatore: **-2** istruisce di generare codice per Python 2.x e **--embed** indica di integrare il metodo main, quindi il file **app.pyx** avrà il proprio metodo main e l'esecuzione inizierà da qui. Se non specificato dovrete lanciare manualmente il runtime Python e importare il modulo, pena un crash! Immaginate di voler usare C per controllare quando viene avviata la vostra app-server. Potete quindi incapsulare la chiamata **app.run()** all'interno del proprio metodo che chiamerete poi da C. Potete farlo definendo ancora una volta la funzione come 'cdef' ma utilizzando anche la keyword 'public':

```
cdef public void startFlaskApp():
        app.run()
```

Il tipo di ritorno è void ma soprattutto la keyword 'public' farà sì che il compilatore Cython generi un header app.h che avrà il prototipo di tale funzione dichiarato per voi, così da poterla richiamare direttamente da C come fosse semplicemente un'altra funzione C, e tutto quello che dovrete fare è includere l'header. Dal momento che avrete il vostro metodo main all'interno di un programma C e chiamerete Cython per lanciare la webapp dovrete scrivere un piccolo boilerplate prima di fare alcunché:

```
#include <Python.h>

int main (int argc, char **argv) {
        printf("Inizializzo runtime Python...\n");
        Py_Initialize();

        // qui le chiamate python
        printf("Termino...\n");
        Py_Finalize();
        return 0;
}
```

Quindi all'interno della funzione main dovete inizializzare il runtime Python e ripulirlo al termine; tra queste due chiamate potete utilizzare codice Python, all'esterno otterrete un segfault. Dal momento che avete dichiarato **startFlaskApp**

› Ecco nella schermata l'esecuzione della versione Cythonizzata di fibonacci con il corrispondente codice in ViM

cdef public potete includere l'header **app.h** dopo l'inclusione di **Python.h** e chiamare direttamente la funzione Cython:

```
// assicuratevi di includere prima 'Python.h'!
#include <Python.h>
#include "app.h"

int main (int argc, char **argv) {
        printf ("Inizializzo runtime Python...\n");
        Py_Initialize ();
        initapp(); // equivalente alla parte python 'import app' di 'app.h'
        startFlaskApp(); // chiamo la funzione
        printf ("Termino...\n");
        Py_Finalize ();
        return 0;
}
```

Compilate il tutto:

```
$ cython -2 app.pyx -o app.c
$ gcc -g -O2 -o app main.c app.c `python-config --includes --ldflags`
```

Quindi lanciate **./app**. Ora avete un'applicazione flask in esecuzione da un programma C! La prossima cosa è definire una funzione che ritorna semplicemente alcuni dati causali in una struttura C. Forse una stringa fissa e un contatore, giusto per semplicità: assumiamo che sia un record in memoria per qualche genere di sistema, dato che il principio è identico.

```
struct data {
        char name[10];
        int value;
};
```

Questa è la definizione della struttura e ora vi serve una funzione che potete chiamare per ottenere un nuovo record.

```
static struct data * record = NULL;
struct data * getData (void)
{
        static int callCount = 1;
        if (!record)
                record = malloc(sizeof(struct data));
        memset (record, 0, sizeof(*record));
        strncpy(record->name, "test", sizeof(record->name));
        record->value = callCount;
        callCount++;
        return record;
}
```

Questa funzione compilerà semplicemente il record per poi servirlo elegantemente con Python come oggetto JSON. Il modo migliore per farlo è mettere la definizione della struttura in un header assieme al prototipo della funzione per il recupero dei dati. Questo è importante, dal momento che Cython compila in C e deve includere un header nell'app C in modo che tutto fili liscio.

```
#ifndef DATA_H
#define DATA_H
struct data {
        char name[10];
        int value;
};
extern struct data * getData (void);
#endif //DATA_H
```

All'interno del codice Cython dovete dichiararlo nuovamente così che il compilatore Cython sappia come gestire correttamente i tipi; in caso contrario non saprà nulla riguardo la struttura o la funzione e genererà un errore.

```
cdef extern from "data.h":
        struct data:
                char name[10]
                int value
        data * getData ()
```

Come potete vedere, è codice con cui avete familiarità ma è noioso riscriverlo ancora. Esistono dei progetti (per esempio auto-generazione di pxd **http://bit.ly/LXFapg**) per generarlo automaticamente ma non sono ancora mature. Il prossimo passo che vi serve è una funzione per incapsulare la chiamata nel codice C per ritornare un dizionario della struttura restituita.

```
cpdef getBackendData():
        cdef data * resp = getData ()
        return {'name': resp.name, 'value': resp.value}
```

In questa funzione è stata usata la keyword **cpdef**, che indica la possibilità di chiamare la funzione da codice Python o C e il corpo può sfruttare l'estensione Cython per utilizzare i tipi nativi. Se chiamerete la funzione da C il tipo restituito è sempre PyObject (un tipo di oggetto interno di Python). Il corpo chiama direttamente **getData** che avete definito nel codice C e converte la struttura in un dizionario Python. Infine, per terminare il tutto dovete integrare la route di Flask, formattare in JSON e servire la risposta!

```
cironattia@mercurio-ubuntu:~/cython$ make
cython -2 flaskApp.pyx -o flaskApp.c
gcc -g -O2 -o app main.c flaskApp.c `python-config --includes --ldflags`

cironattia@mercurio-ubuntu:~/cython$ ./app
Inizializzo runtime Python...
 * Running on http://127.0.0.1:5000/
127.0.0.1 - - [29/Dec/2014 14:51:18] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Dec/2014 14:51:19] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Dec/2014 14:51:20] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Dec/2014 14:51:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Dec/2014 14:51:22] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Dec/2014 14:51:23] "GET / HTTP/1.1" 200 -
```

```
cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 1
}cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 2
}cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 3
}cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 4
}cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 5
}cironattia@mercurio-ubuntu:~/cython$ curl -X GET "http://localhost:5000"
{
  "name": "test",
  "value": 6
}cironattia@mercurio-ubuntu:~/cython$
```

› **Esecuzione dell'app Flask e utilizzo di curl per accedere alla risorsa json**

Alternative a Cython

Questo tipo di binding può essere ottenuto anche con **Swig**. Cython brilla nella tipizzazione e nella implementazione diretta che offre. Per esempio, in Swig dovete scrivere alcune funzioni per l'incapsulamento dei tipi per impostarli tra i diversi linguaggi. Dovrete anche scrivere funzioni per l'incapsulamento di strutture dati più complesse. Cython lo fa per voi dal momento che riconosce i tipi! Non è per dire che Swig sia male, ma è molto generalizzato dal momento che supporta così tanti linguaggi come Java, Go, ecc. Cython è pensato unicamente per Python e può basarsi su basi molto più definite. Altri progetti notevoli in questo ambito sono **Parakeet**, che usa LLVM per compilare il codice just-in-time con l'uso dei decoratori nel codice Python per velocizzare il codice, e Nuitka, un compilatore da Python a C++ che prova a tipizzare staticamente il vostro programma Python a compile-time analogamente a PyPy.

```
from flask import Flask, jsonify

app = Flask ('myapp')
cpdef getBackendData():
        cdef data * resp = getData ()
        return {'name': resp.name, 'value': resp.value}
@app.route("/")
def index ():
        return jsonify (getBackendData ()) cdef public void
startFlaskApp ():
        app.run ()def index ():
   return jsonify (getBackendData ())
cdef public void startFlaskApp ():
   app.run ()
```

Ricapitolando, quello che avete finora è un singolo file C contenente il metodo **main** per inizializzare il runtime Python e lanciare **flaskApp**, assieme a una funzione per recuperare dei dati, un header e un file Cython (*.pyx) che contiene il codice per la app flask che serve le risposte. Per compilare e collegare il tutto potete scrivere un semplice **Makefile**:

```
all:
        cython -2 app.pyx -o app.c
        gcc -g -O2 -o app main.c app.c `python-config
--includes --ldflags`
```

Lanciate l'app:

```
$ ./example
Inizializzo runtime Python...
 * Running on http://127.0.0.1:5000/
```

Utilizzando curl per accedere alla route flask notate come ogni richiesta aggiornerà l'oggetto JSON:

```
$ curl -X GET "http://localhost:5000"
{
        "name": "test",
        "value": 1
}
$ curl -X GET "http://localhost:5000"
{
        "name": "test",
        "value": 2
}
```

Ed ecco fatto: avete chiamato codice Python da C e chiamato codice C da Cython! Da questo punto in poi potete fare moltissime cose, per esempio mappare risorse RESTful in classi C++. Cython sta migliorando a vista d'occhio riguardo il supporto per C++: potete anche definire template e sovraccaricare gli operatori su classi e Cython conosce come gestire il tutto. Ci sono alcune attenzioni da avere riguardo a Cython prima di cominciare a scrivere codice. Dal momento che state mescolando linguaggi, il debug può confondervi. Mantenere le interfacce semplici è la chiave di volta, e la gestione delle eccezioni da Python a C++ può portarvi fuori strada dal momento che queste saranno modificate in base al linguaggio dal quale state partendo. Non potrete magicamente compilare un intero modulo Python come libreria condivisa, poiché Cython non segue gli import. Per gestire più file Cython dovete usare una direttiva include analogamente agli include C/C++, così che il codice venga integrato in-place, questo per conformità con il sistema di moduli di Python. Per esempio, potete notare l'uso frequente di **--embed** nel codice, o la definizione di un metodo **main** in Cython per poter creare un modulo:

```
def myfunction():
        return 'Hello World'
```

e lo potete compilare con:

```
cython -2 mycode.pyx -o mycode.c
gcc -g -O2 -shared -o mycode.so mycode.c `python-config
--includes --ldflags`
```

Ora avete una libreria condivisa e potete includerla come qualsiasi altro modulo Python:

```
>>> import mycode
>>> mycode.myfunction ()
'Ciao mondo!'
```

Assicuratevi solamente che il nome della libreria condivisa non sia lo stesso del file con il codice Cython a meno dell'estensione. Cython non finisce qui, offre un'interfaccia estremamente pulita per lavorare con codice C++ incluse classi e template, e con la gestione delle eccezioni in Python e C++ c'è anche un eccellente supporto a GDB (GNU Debugger). Cython è stato usato per grossi sistemi economici multi-processo, che sfruttano a fondo database personalizzati in memoria condivisa, così da eliminare l'overhead del dialogo con il database. Il rovescio della medaglia è che il debug e l'analisi del contenuto corrente delle tabelle è solitamente difficile, oltre a problemi noti come la crescita della memoria condivisa. Utilizzando Cython e Flask è stato però possibile creare una dashboard utilizzando Python e AngularJS per costruire un visualizzatore realtime di database per le transazioni. C'è di più: è stato possibile il semplice riuso delle API C++ per far funzionare il tutto direttamente in Cython, così da abilitare la scrittura trasparente di codice Python e C++. Nelle grosse istituzioni finanziarie avrete sempre una risposta migliore se sfruttate l'estensione e il riuso, invece di nuovo codice o API. Questa tecnica può essere applicata molto facilmente anche ad altre aree, anche nella scrittura di nuove applicazioni C/C++. Perché non scrivere wrapper per utilizzare il logger di Python o l'analizzatore di configurazione dall'interno di applicazioni C/C++ così da non dover reinventare la ruota per fare una cosa così semplice? Se volete una dimostrazione di alcune di queste tecniche leggete **Learning Cython Programming** di Philip Herron (disponibile su Amazon: **http://bit.ly/LearningCythonIt** e **https://github.com/redbrain/cython-book**). Anche se Cython è nato principalmente per avvantaggiarsi dei tipi nativi in computazioni matematiche a elevata performance, potete utilizzarlo in numerose altre aree. LXP

Lavorare con Git

Lo staff di Linux Pro vi spiegherà le nomenclature di base di **Git** in modo che anche voi possiate padroneggiare i fork e le fusioni a tre vie



Git è un tool di controllo delle versioni e di distribuzione del codice sorgente, per diversi aspetti assomiglia più a un filesystem che agli altri sistemi di controllo delle versioni, come **CVS** e **Mercurial**. Il codice è salvato in un repository (locale o remoto). Gli utenti con i permessi adeguati possono copiare il repository nella loro directory locale, e avere accesso non solo al ramo principale del codice sorgente, ma a tutte le ramificazioni e a tutte le loro storie. Possono modificare e inviare le modifiche al loro repository locale e se necessario fare una richiesta per incorporare le modifiche al progetto. Poco dopo la sua uscita Git ha iniziato ad hostarsi da solo, anche il suo codice sorgente è gestito tramite un repository Git. Quando clonate o inizializzate un repository, a un primo sguardo vedrete soltanto lo stato attuale (o per dirla nel linguaggio Git il ramo principale) del progetto. Ma se guardate più attentamente potrete notare una directory nascosta chiamata **.git/**, dove risiede tutta la magia. Questa cartella è il posto nel quale vengono salvati i file di configurazione e gli script, contiene inoltre gli indici (**.git/index**) e gli oggetti database (**.git/objects/**). L'indice è un'area di staging e lavora con lo stato della directory in uso, tracciando le modifiche in attesa del commit. Date un'occhiata al box **Oggetti database** per maggiori dettagli). Sebbene Git sia nato per gestire il codice sorgente, è in grado anche di gestire qualunque progetto che coinvolga delle modifiche ai file. Forse state Photoshoppando (scusate, Gimpando) fuori i suoceri dalle vostro foto, o sperimentando diverse trame per il romanzo su cui state lavorando, in entrambi i casi Git può aiutarvi. Con questo secondo esempio in mente, vediamo come creare il nostro repository Git personale. Avremo bisogno di installare Git ovviamente; sicuramente potete trovarlo nei repository della vostra distribuzione.

```
        new file:   capitolo1
        new file:   capitolo2

light@gonzo:~/demogit$ git commit -m 'Commit iniziale, maggiordomo'
[master (root-commit) 7c20a7f] Commit iniziale, maggiordomo
 2 files changed, 2 insertions(+)
 create mode 100644 capitolo1
 create mode 100644 capitolo2
light@gonzo:~/demogit$ git status
Sul branch master
nothing to commit, working directory clean
light@gonzo:~/demogit$ git log
commit 7c20a7f[illegible]
Author: Light <matteo@gonzo.it>
Date:   Tue Dec 16 11:05:06 2014 +0100

    Commit iniziale, maggiordomo
light@gonzo:~/demogit$ git show 7c20a7
commit 7c20a7f[illegible]
Author: Light <matteo@gonzo.it>
Date:   Tue Dec 16 11:05:06 2014 +0100

    Commit iniziale, maggiordomo

diff --git a/capitolo1 b/capitolo1
new file mode 100644
index 0000000..e719e38
--- /dev/null
+++ b/capitolo1
@@ -0,0 +1 @@
+È stato il maggiordomo
diff --git a/capitolo2 b/capitolo2
```

› **I comandi git {status,log,show} sono utili per vedere a che punto siete con i commit e da dove venite**

Che i commit abbiano inizio

La prima cosa che dovrete fare è dire a Git il vostro nome e la vostra email, queste informazioni saranno visibili se inizierete a committare dati su altri progetti.

```
git config --global user.name "Vostro nome"
git config --global user.email vostra@email.it
```

Ora creiamo un progetto iniziale con un paio di file di testo, e poi mettiamo la cartella del progetto sotto l'ala protettrice di Git:

```
mkdir ~/demogit
cd ~/demogit
echo "È stato il maggiordomo" > capitolo1
echo "Da fare: scrivere emozionante colpo di scena" > capitolo2
git init
```

L'ultimo comando semplicemente crea la directory **.git/**, Git non inizierà a prendersi cura dei nostri file finché non lanceremo questo comando:

```
git add .
```

Ora tutti i file nella directory di lavoro sono salvati nel file di indice di cui parlavamo prima. Questo file salva ogni modifica in attesa del commit, che faremo tra poco. Per prima cosa eseguiamo questo comando per vedere che tutto funzioni correttamente:

```
git status
```

Da notare che il comando **status** mostra soltanto il ramo principale, stiamo preparando il nostro primo commit e che questo consisterà nella creazione dei nostri file **capitolo1** e **capitolo2**. Per eseguire il nostro primo commit facciamo:

```
git commit -m 'Commit iniziale, maggiordomo'
```

Il commit necessita sempre di una descrizione, in questo modo la gente può farvi sapere quando fa danni. Se non ne specificate

Oggetti database

Il database degli oggetti salva tutto ciò che esiste e che è esistito all'interno del progetto. Può solo crescere, gli oggetti, una volta committati, non vengono mai cancellati altrimenti non riusciremmo ad avere una storia completa. A ogni oggetto viene assegnato un ID univoco lungo 40 caratteri esadecimali (160 bit), generato da una hash SHA1 del suo contenuto e di altri dati importanti. Gli oggetti sono salvati in base al loro id, rendendoli accessibili quindi in base al loro contenuto. I primi due caratteri dell'id sono utilizzati per la sottocartella in **.git/objects**, mentre gli altri 38 costituiscono il nome del file. Il modello degli oggetti di Git prevede quattro entità distinte:

» **Blobs (oggetti binari di grandi dimensioni)** Il contenuto dei file, dati amorfi. È importante notare che un blob non è la stessa cosa di un file, dato che non è associato al nome del file, a una data o a dei permessi. Blob identici avranno lo stesso id, quindi non viene sprecato spazio salvando dei duplicati, anche se il nome del file associato ai dati del blob cambia.

» **Trees (alberi)** Sono simili alle normali directory, contengono una lista di nomi di file e altri metadati oltre all'associazione con gli id dei blob (o di altri alberi). Quindi un tree contiene l'immagine di un dato stato del progetto.

» **Commits** Connettono gli alberi in modo da formare una storia. Contengono un puntatore a un albero che rappresenta lo stato del repository dopo che sono state fatte le modifiche. Contengono inoltre referenze a commit precedenti; in questo modo è facile invertire il processo. Inoltre, salvano informazioni relative a chi ha eseguito il commit, quando è stato fatto e una versione leggibile delle modifiche.

» **Tags** Puntano agli oggetti commit, vi permettono di dare un nome capibile a un commit. Potete anche aggiungere un messaggio descrittivo.

una con l'opzione **-m** si aprirà un editor per farvela inserire. Come potete vedere nell'immagine, il nostro commit iniziale ha un id che comincia con **7c20a7** (il vostro potrebbe essere diverso visto che dipende dall'ora e dai dettagli dell'utente). Normalmente sono necessarie soltanto le prime cifre dell'id, dato che le possibilità di id doppi sono minime. L'id completo può essere visualizzato tramite il comando **git log**. Oppure potete vedere tutti i dettagli con:

```
git show 7c20a7
```

Se date un'occhiata alla directory **.git/objects**, dovreste vedere due directory chiamate **info** e **pack** oltre a quattro directory di due lettere. Queste ultime quattro directory dovrebbero contenere un file ognuna, il nostro progetto è formato da quattro oggetti: due blob (con id **e719e3** e **dbd4b1**), un tree (**8c1d08**) e un commit (**7c20a7**). Ora proseguiamo con la nostra macabra trama, aggiungendo alcuni dettagli sul presunto crimine commesso dal maggiordomo. Aggiungiamo una riga al **capitolo1** scrivendo:

```
echo "con il fucile, nello studio" >> capitolo1
```

Ovviamente, potete utilizzare il vostro editor di testo preferito per fare le modifiche. Lanciare **git status** ci dirà che queste modifiche non sono in coda nel commit, dobbiamo lanciare nuovamente il comando **git add** sui file, o utilizzare l'opzione **-a** per fare commit di tutti i file presenti nell'indice. Non dimenticatevi di aggiungere una descrizione:

```
git commit -a -m 'dettagli, arma utilizzata e scena del crimine'
```

Git ci informa che abbiamo modificato un file con un inserimento, e come prima possiamo avere maggiori informazioni utilizzando **git log** e **git show**. Abbiamo anche aggiunto altri tre oggetti. Un nuovo blob contenente le due righe del **capitolo1** (Git non utilizza inizialmente la compressione delta), un nuovo tree e il nostro nuovo commit (id **1e407a**). Il diagramma che potete vedere a destra mostra cosa è successo alla struttura degli oggetti nei due commit. Da notare che il nostro secondo commit ha una referenza al padre (il commit iniziale con id **7c20a7**), e che entrambi gli alberi puntano allo stesso blob con id **dbd4b1**, cioè il **capitolo2** dato che non ha subìto modifiche. Se decidiamo che l'ultimo commit eseguito non ci piace, possiamo riportare indietro la directory corrente utilizzando il comando:

```
git checkout 7c20a7
```

Git ci avviserà che siamo entrati nella modalità (dal nome non molto rassicurante) 'detached HEAD' (letteralmente testa staccata). Possiamo dire di essere tornati indietro nel tempo. Potete verificare ciò che è successo esaminando il **capitolo1**, vedrete che avrà una riga soltanto e non ci sarà traccia della seconda riga inserita poco fa.

Potete fare modifiche senza aver paura di rovinare il progetto: a differenza della fantascienza, Git ha dei metodi ben definiti per modificare il passato senza compromettere il futuro, come vedremo tra un momento. Per prima cosa, torniamo al presente con:

```
git checkout master
```

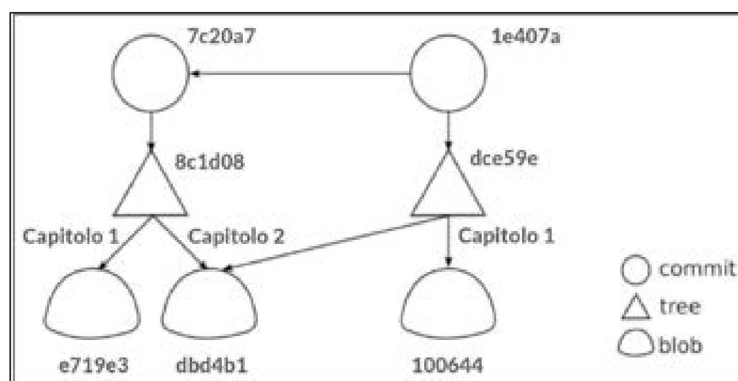
Rami e narratori inaffidabili

Nel linguaggio di Git i progetti sono divisi in rami (*branches*), che sono linee di sviluppo indipendenti. Il processo di creare un nuovo ramo è conosciuto come 'forking'. È buona norma isolare ogni modifica che si fa in un progetto all'interno del proprio ramo, principalmente per evitare che queste modifiche vadano a rompere o inquinare il codice di base, conosciuto come *master branch* (ramo principale). Vediamo ora come creare un nuovo ramo per fare esperimenti con la trama, e dire a Git che vogliamo lavorare con essa. I rami sono semplicemente dei puntatori a dei commit, e possiamo allo stesso modo, utilizzando **git checkout**:

```
git branch colpo-di-scena
git checkout colpo-di-scena
```

A differenza dei commit, comunque, fare il checkout di un ramo vi permetterà di salvare le modifiche nella storia del progetto. Facciamolo con un nuovo secondo capitolo:

```
echo "Sono stato io, il vostro narratore, che ha commesso il vile atto. Ho incolpato il maggiordomo per gelosia, e se anche, tu, caro lettore, puoi vedere la mia confessione in queste pagine, nessun altro potrà farlo. In verità, sono un vile della peggior specie" > capitolo2
```



› **Dopo un paio di commit, il database degli oggetti diventa un po' complicato. Fortunatamente git gestisce tutta la complessità per noi**

Tip

Il nome Git è un altro esempio di progetto autoreferenziale di Linus Torvalds. Pensava che l'espressione volgare inglese che significa "persona sgradevole" lo descrivesse bene.

Che colpo di scena, siamo certi che non ve lo sareste mai aspettati, degno di un romanzo di Poe. Ora dobbiamo aggiungere questo file all'area di staging per poter fare commit delle modifiche, possiamo farlo con una sola riga di codice:

```
git commit -a -m "Confessione a sorpresa dall'autore"
```

Se a questo punto decidiamo che ci piace questo ramo, possiamo farlo diventare quello principale. Questa operazione è conosciuta come *merging* (fusione), in questo caso semplice e lineare (c'è un percorso diretto dal ramo principale al nostro ramo colpo-di-scena) la fusione non è altro che un avanti veloce. Prima di fondere il tutto dobbiamo spostarci nel ramo principale e poi eseguire la fusione:

```
git checkout master
git merge colpo-di-scena
```

Il nostro ramo colpo-di-scena continua a vivere ma possiamo effettivamente eliminarlo con **git branch -d colpo-di-scena** dato che non ci serve più.

Fusione a tre vie

Ora, immaginiamo di essere tornati nel ramo principale e di aver modificato il secondo capitolo, di aver deciso poi che il colpo di scena del narratore fosse migliore, e quindi avessimo deciso di portare avanti quella parte. La fusione sarebbe stata un po' più complicata (non c'è più un percorso lineare tra i due rami), Git ci aiuta a risolvere questi problemi con le fusioni a tre vie. Il nome deriva dal fatto che la soluzione arriva considerando i tre commit in grassetto nel diagramma qui sotto: i due rami e il loro antenato comune.

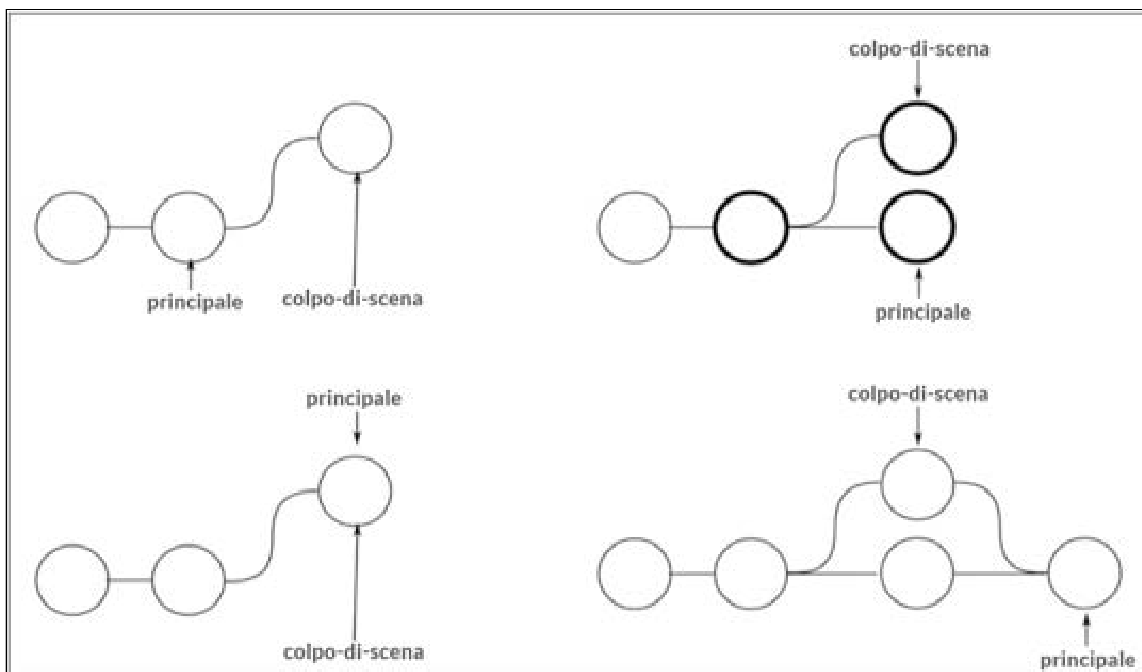
Il comando è lo stesso, ma questa volta Git ci dirà che c'è un conflitto nella fusione di **capitolo2**. In realtà, se guardate al contenuto di questo file vedrete l'insieme delle due versioni.

Per risolvere il conflitto, dobbiamo fare il **checkout** soltanto della versione presente in colpo-di-scena del nostro **capitolo2**, a quel punto la fusione può proseguire senza ulteriori intoppi:

```
git checkout colpo-di-scena capitolo2
git commit -m "Fusione con il ramo colpo-di-scena"
```

Git è in grado di fondere in autonomia i file se differiscono in modo semplice da capire, per esempio se uno dei due contiene un paragrafo aggiuntivo o un pezzo di codice, esistono anche alcune strategie preprogrammate (chiamate *ours* (nostre), *theirs* (loro) e *both* (entrambi)) per fusioni semplificate.

Nella maggior parte dei casi la fusione è più complicata e richiede l'intervento dell'utente. In casi ancora più complicati può coinvolgere diversi rami, Git deve ricorrere a ciò che è conosciuto come tre vie ricorsivo. Dopo quest'ultima informazione concludiamo il nostro tutorial. Sappiamo di aver semplicemente grattato la superficie sulle potenzialità di questo tool per la gestione delle versioni, ma speriamo di avervi incuriosito. Parleremo in modo più approfondito di Git in un prossimo articolo ma per il momento, buon divertimento. LXP



› Una fusione fast-forward (avanti veloce) (in alto a sinistra, con il risultato in basso a sinistra) e una fusione a tre vie (three-way merge) sulla destra. Nel secondo caso il risultato sarà un nuovo commit

Un GitHub anche un po' vostro

GitHub (**https://github.com**) è un repository Web dove chiunque può salvare i propri progetti gratuitamente, a condizione che i progetti siano di dominio pubblico. È disponibile anche un servizio a pagamento per chi vuole tenere il proprio codice per se stesso. Oltre a essere un repository Git, GitHub offre agli utenti anche altri tool per gestire i propri progetti, inclusa la wiki e un servizio di bug tracking. Una volta registrati in GitHub potete caricare un repository locale sul loro server tramite HTTPS o SSH. Il primo richiede l'accesso tramite password mentre il secondo vi permette di utilizzare le chiavi SSH. Una volta inizializzato il vostro repository e caricati i primi file (è obbligatorio includere almeno un file **README.md** che descriva il progetto) potete aggiungere l'indirizzo remoto in questo modo

```
git remote add origin https://github.com/nomeutente/nomerepository
```

e quando siete pronti a inviare il vostro ramo principale vi basterà lanciare:

```
git push -u origin master
```

Potete dare accesso al vostro repository GitHub a tutte le persone di cui vi fidate, e loro potranno aggiungere i loro rami e contributi. Un deposito centrale gestito in questo modo viene spesso definito "bare repository" (repository nudo), dato che nulla viene committato direttamente, tutti i contributi provengono dai repository locali vostri e dei vostri aiutanti.

IN EDICOLA OGNI 14 GIORNI

+ FACILE
TANTI GIOCHI
PER TUTTI

TANTI LIVELLI E DIVERSI GRADI DI DIFFICOLTÀ PER DIVERTIRTI GIOCANDO

L'ALLENA-MENTE CHE DIVERTE E APPASSIONA!

LOGiKA

CI VEDIAMO IN EDICOLA OGNI DUE MARTEDÌ

PAROLIAMONE
I PIÚ BEI GIOCHI
DI PAROLE,
I PIÚ APPASSIONANTI

LOGIKA JUNIOR
I MIGLIORI ENIGMI
PER I CAMPIONI
DI DOMANI

AGUZZA LA VISTA
TANTI NUOVI GIOCHI PER
ALLENARE IL VOSTRO...
COLPO D'OCCHIO!

TEST D'INTELLIGENZA
ALLENA IL TUO CERVELLO
IN UNA GARA
CONTRO IL TEMPO

Paroliamone
I più bei giochi con le parole!

LOGiKA JUNIOR
I giochi logici per i campioni di domani!

NUOVA LOGIKA
Prova un metodo di gioco tutto nuovo!

GIOCHI A TEMPO
Trova i due cubi identici in meno di 2 minuti!
pag. 11

GRIGLIE LOGICHE
Segui gli indizi e risolvi l'enigma...
pag. 16

I N S T R A D A T E

A pagina 22 trovi le regole del gioco
Adibita ad un particolare servizio - Dura un attimo - Fa accomodare i pazienti su poltrone reclinabili - Il romanziere Fleming - La risposta dell'indeciso - Madri di bardotti - Proverbialmente lo sono i pesci - Triplice quella di Francia, Gran Bretagna e Russia.

LogiKAngourou
Calcola velocemente il
QUOZIENTE D'INTELLIGENZA

"Se è verde o si muove, è biologia. Se puzza, è chimica. Se non funziona, è fisica."

GIOCHIAMO A

IN CRESCENDO!

LA REGOLA È FACILE: risolvi lo schema inserendo nelle righe le parole corrispondenti, tenendo presente che ogni termine si ottiene da quello della riga sovrastante aggiungendogli una lettera e anagrammando il tutto (es.: caro + c = rocca). Le lettere aggiunte vanno inserite nelle caselle evidenziate. Le definizioni delle otto parole intermedie sono riportate sulla sotto in ordine alfabetico.

Avverse, sfavorevoli - È prodotta dalle api - In molte scuole c'è quella di Natale - L'arte di parlare bene - La sigla di Caserta - Quaranta in un mazzo da briscola - Sono cavità... sulla Luna - Un gusto dei succhi di frutta

E

I N C R O S T A R E

METTITI ALLA PROVA CON I GIOCHI DI LOGICA, QUIZ, SUDOKU, GRATTACIELI, CRUCIPIXEL... E TANTI ALTRI!



L'eco dei LUG

I Lug

I LUG rappresentano da sempre il punto di riferimento per chiunque voglia conoscere GNU/Linux. Ogni mese dedicheremo loro questo spazio per la comunicazione di nuovi progetti e appuntamenti. Se hai qualcosa da segnalarci scrivi a ecodeilug@linuxpro.it

ABRUZZO

AnxaLUG - Lanciano
www.anxalug.org
Il Pinguino - Teramo
Non disponibile
MarsicaLUG - Marsica
www.marsicalug.it
OpenLUG - L'Aquila
Non disponibile
Pescara LUG
www.pescaralug.org
Pineto LUG
www.pinetolug.org
Pollinux LUG - Pollutri
Non disponibile
SSVLUG - San Salvo, Vasto, Termoli
www.ssvlug.org
SulmonaLUG
http://sulmonalug.it
TeateLUG - Chieti
Non disponibile
TeLUG - Teramo
www.telug.it
User Group Valle Roveto
http://linuxvalley-os4.blogspot.com/

BASILICATA

Basilicata LUG - Potenza e Matera
www.baslug.org

CALABRIA

3BYLug - Trebisacce
www.3bylug.tk
Bogomips - Bisignano
www.blug.it
CastroLUG
http://castrolug.altervista.org
Cosenza Hack Laboratory
http://hacklab.cosenzainrete.it/
CSLUG - Cosenza
http://cslug.linux.it
CzLug
Non disponibile
HackLab Catanzaro
http://hacklab.cz
Piana LUG - Piana di Gioia Tauro
Non disponibile
Reggio Calabria LUG
http://rclug.linux.it
Revolutionary Mind
www.revolutionarymind.org
SpixLug - Spezzano Albanese
Non disponibile

CAMPANIA

AFR@Linux LUG
www.afralinux.netsons.org
Afralug - Afragola
www.afralug.com
CasertaLUG
www.casertaglug.org
Hackaserta 81100
www.81100.eu.org
HackMeetNaples Napoli HackLab
www1.autistici.org/hmn
IGLUG - Napoli e provincia
www.iglug.org
IRLUG - Irpinia
www.irlug.it
LUG-Ischia
www.lug-ischia.org
NALUG - Napoli
www.nalug.net
Neapolis Hacklab
www.officina99.org/hacklab.html
Padulug - Paduli (BN)
http://linux.paduli.com
SCALUG - Scafati (SA)
http://xoomer.alice.it/scalug/
Tuxway.org - Provincia di Napoli
www.tuxway.org
ValLug - Vallo Linux User Group
www.valug.it
XALUG - Salerno
http://xalug.tuxlab.org

EMILIA ROMAGNA

ALFLUG - Alfonsine
www.alflug.it
Borgotaro LUG - Val Taro
http://btlug.it/
ConoscereLinux - Modena
www.conoscerelinux.it
ERLUG
http://erlug.linux.it
Ferrara LUG
www.ferrara.linux.it
FoLUG - Forlì
http://folug.linux.it
ImoLUG - Imola
www.imolug.org
LUGPiacenza
www.lugpiacenza.org
PANLUG - Vignola
Non disponibile
PLUG - Parma
http://parma.linux.it
RavennaLUG
www.ravennalug.org
RELug - Reggio Emilia e provincia
http://relug.linux.it
RiminiLug
www.riminilug.it
S.P.R.I.Te
http://sprite.csr.unibo.it
UIELinux - Valle del Rubicone
www.uielinux.org

FRIULI VENEZIA GIULIA

GOLUG - Gorizia
www.golug.it
IGLU - Udine
http://iglu.cc.uniud.it
LUG Pordenone
www.pordenone.linux.it
LugTrieste
http://trieste.linux.it
LUG [A] [L] [P] - Aquileia
www.alproject.org

LAZIO

CiLUG - Frosinone
www.cilug.org
CLUG - Cassino
http://cassino.linux.it/
GioveLUG - Terracina
www.giovelug.org
La Sapienza LUG
www.lslug.org
Latina LUG
www.llg.it
LUG Privernum Volsca - Priverno (LT)
www.pvlug.org
LUGRoma
www.lugroma.org
LUGRoma 3
www.lugroma3.org
TorLUG - Università Tor Vergata - Roma
http://lug.uniroma2.it/
V.I.S.C.O.S.A. - Ciampino
www.viscosa.org

LIGURIA

Genuense Lug - Genova e d'intorni
http://genova.linux.it
GinLug - Genova Sampierdarena
www.sennaweb.org
Govonis GNU/LUG - Provincia di Savona
www.govonis.org
SLIMP - Software Libero Imperia
http://slimp.it/
TLug-TSL - Tigullio Ligure
http://tlug.linux.it/

LOMBARDIA

BGLug - Bergamo e provincia
www.bglug.it
BGLug Valle Seriana - Valle Seriana
http://bglugvs.web3king.com/
GL-Como - Como
www.gl-como.it
GLUX - Lecco e provincia
www.lecco.linux.it
GULLP - Gruppo Utenti Linux Lonate Pozzolo
www.gullp.it
IspraLUG - Ispra
http://ispralug.eu/
LIFO - Varese
www.lifolab.org
LIFOS - Cinisello Balsamo
www.lifos.org
Linux Var - Varese
www.linuxvar.it
LoLug - Lodi e provincia
www.lolug.org
Lug Bocconi - Milano
www.lug-bocconi.org
LugBS - Brescia e provincia
http://lugbs.linux.it/
Lug Castegnato - Castegnato
www.kenparker.eu/LugCastegnato
LugCR - Cremona e provincia
www.lugcr.it
Lug Crema - Crema
http://filibusta.crema.unimi.it/
LUGDucale - Vigevano
www.lugducale.it
LugMan - Mantova e provincia
www.lugman.org

L'eco dei LUG

LugOB - Cologne e ovest bresciano
www.lugob.org
MoBLUG - Monza e Brianza
www.bubblesfactory.it
OpenLabs - Milano
www.openlabs.it
POuL - Milano
www.poul.org
TiLug - Pavia
http://pavia.linux.it
VigLug - Vignate, Milano Est - Adda Martesana
www.viglug.org

MARCHE

Ascolinux LUG/FSUG Ascoli
http://marche.linux.it/ascoli/
CameLUG - Camerino
www.camelug.it
CMlug
www.cmlug.org
Egloo
www.egloo.org
FanoLUG
www.fanolug.org
Fermo LUG
www.linuxfm.org/fermolug/
GLM - Macerata
www.gruppolinuxmc.it/start/index.php
LUG Ancona
www.egloo.org
LUG Jesi
www.lugjesi.net
LUG Marche
http://marche.linux.it
PDP Free Software User Group
http://pdp.linux.it
SenaLug - Senigallia
www.lug.senigallia.biz

MOLISE

Campobasso LUG
http://cb.linux.it/
FrenterLUG - Larino
non disponibile
SmaLUG - San Martino
www.smalug.org

PIEMONTE

ABC Lug - Alba/Bra/Carmagnola
http://abc.linux.it/
AlLug - Alessandria e provincia
www.allug.it
BiLUG - Provincia di Biella
http://www.bilug.it
FASoLi - Alessandria e provincia
http://softwarelibero.al.it/
Gallug - Galliate
www.gallug.it
GlugTO - Torino e provincia
www.torino.linux.it
IvLug - Ivrea Linux User Group
www.ivlug.it
Linox Novara
www.linoxnovara.org
SLIP - Pinerolo
http://pinerolo.linux.it/
ValSusinux - Val Susa e Val Sangone
www.valsusinux.it

PUGLIA

BriLUG - Brindisi
www.brilug.it
CapitanLUG - Capitanata
www.capitanlug.it
LATLUG - Latiano Linux User Group
www.latlug.org
LUGargano
www.lugargano.it
LUGBari - Bari e provincia
www.lugbari.org
MurgiaLug - Santeramo in Colle
www.open-pc.eu/index.php/murgialug/
SaLUG! - Salento
http://salug.it
Talug - Taranto
www.talug.it

SARDEGNA

GNUraghe - Oristano
www.gnuraghe.org
GULCh - Cagliari
www.gulch.it
PLUGS - Sassari
www.plugs.it

SICILIA

CefaLug - Cefalù
http://cefalug.linux.it
cLUG - Caltanissetta
www.clug.it
EnnaLUG
www.ennalug.org
FreakNet MediaLab - Catania
www.freaknet.org
Leonforte LUG
http://leonforte.linux.it
LUG Catania
www.catania.linux.it
LUGSR - Siracusa
www.siracusa.linux.it
MELUG - Messina
non disponibile
Norp LUG - Noto, Pachino, Rosolini
non disponibile
PALUG - Palermo
http://palermo.linux.it
RgLUG - Ragusa e provincia
http://ragusa.linux.it
VPLUG Linux Planet - Provincia Caltanisetta
www.vplug.it
SputniX - Palermo
www.sputnix.it

TOSCANA

ACROS - Versilia, Lucca, Massa Carrara
www.lug-acros.org
Elbalinux
non disponibile
ElsaGLUG - Val d'Elsa
www.elsaglug.org
FLUG - Firenze
www.firenze.linux.it
GOLEM - Empoli, Valdelsa
http://golem.linux.it
GroLUG - Grosseto
www.grolug.org
G.U.L.LI - Livorno
www.livorno.linux.it
GulP! Piombino
http://gulp.perlmonk.org
GULP Pisa
www.gulp.linux.it
GuruAtWork - Grosseto e provincia
www.guruatwork.com
IPIOS - Bibbiena e valle del Casentino
www.ipios.org
Lucca LUG
http://luccalug.it
L.U.G.A.R - Arezzo
non disponibile
PLUG - Prato e provincia
www.prato.linux.it
PtLug - Pistoia e provincia
www.ptlug.org
SLUG - Siena e provincia
www.siena.linux.it

TRENTINO ALTO ADIGE

LinuxTrent - Trento
http://linuxtrent.it
LugBz - Bolzano
www.lugbz.org

UMBRIA

OrvietoLUG
www.orvietolug.it
LUG Perugia
www.perugiagnulug.org
TerniLUG
www.ternignulug.org

VALLE D'AOSTA

SLAG - Aosta
www.slag.it

VENETO

0421ug - Provincia di Venezia
www.0421ug.org
BLUG - Belluno
http://belluno.linux.it
Faber Libertatis - Padova
http://faberlibertatis.org
GrappaLUG - Bassano del Grappa
http://grappalug.homelinux.net/
ILC - Informatica Libera Cittadellese - FSUG
http://ilc.pd.it
LegnagoLUG
non disponibile
Linux Ludus - Villafranca (VR)
www.linuxludus.it
LugAnegA
www.luganega.org
LUGSF - San Fidenzio
non disponibile
LUG Vicenza
www.vicenza.linux.it
LugVR - Verona
www.verona.linux.it
MontelLUG - Montebelluna
www.montellug.it
FSUG Padova
www.fsugpadova.org
RoLUG - Rovigo
http://rovigo.linux.it
TVLUG - Treviso
www.tvlug.it
VELug - Venezia
www.velug.it
AViLUG Schio
http://www.avilug.it/doku.php

NAZIONALI

FSUGitalia
www.fsugitalia.org
Gentoo Channel Italia
www.gechi.it
MajaGLUG
www.majaglug.net
SkyLUG
http://tech.groups.yahoo.com/group/skylug/

Guida Software

Ogni mese Linux Pro vi offre i programmi e le distribuzioni più recenti su DVD



Ogni volta che troverete questo simbolo in un articolo, vorrà dire che i file citati si trovano nel DVD allegato alla rivista.

Distro Desktop

Tango Studio 2.2

Uno dei tanti aspetti positivi delle distro Linux è che ne esiste una per qualsiasi cosa. Un po' come le app dei dispositivi mobile. È il caso delle distribuzioni incentrate sul comparto multimediale come **Tango Studio 2.2**. Basato su Debian, questo sistema mette a disposizione tutti gli strumenti più potenti per realizzare, modificare e gestire qualsiasi file multimediale.

Installazione facile e veloce

La versione 2.2 di Tango Studio è stata rilasciata circa sei mesi dopo la precedente release (2.1). Tra le novità, abbiamo l'upgrade del core di Debian a Wheezy 7.5 e l'aggiornamento del Kernel alla versione 3.2.57-realtime. In più, sono state incluse diverse funzioni al desktop, tra cui la ricerca veloce in Synaptic, diversi fix che hanno corretto vari bug, l'aggiunta di ulteriore supporto per audio e video e molto altro ancora. Chi non ha ancora installato la versione principale, potrà scaricare la ISO con il sistema completo. Chi, invece, ha già Tango Studio 2.1, potrà effettuare l'upgrade direttamente da Synaptic o con **apt-get**.
La maggior parte dei pacchetti contenuti nella distro sono liberi, tuttavia ce ne sono un paio proprietari. Stiamo parlando di **Alsa-firmware 1.0.25** e **Wineasio 0.9.0**. Se siete puristi del Software Libero, pertanto, dovrete cercare altre distro.
Il processo d'installazione è in stile Debian e può essere portato a compimento sia attraverso l'installazione grafica (con supporto per il mouse) sia testuale. In entrambi i casi non è per niente complesso, tanto che la prima parte relativa alla configurazione di rete viene gestita automaticamente dal sistema. Tutto quello che dobbiamo fare in questa fase, è aggiungere il nome host e impostare gli utenti con relativa password di amministrazione. Nei vari passaggi che si susseguono, arriviamo comunque a mettere mano a ogni aspetto del sistema. È presente anche uno strumento di partizionamento che consente di gestire a dovere il disco fisso, impostando automaticamente LVM o LVM cifrato. Se però volete procedere manualmente, niente impedisce di farlo. Una volta decisa la configurazione dell'unità, basta dare il via alla scrittura dei dati. Dopo l'installazione base del sistema si passa alla configurazione del gestore pacchetti e alla scelta del bootloader tra GRUB e LILO. Una volta terminato il processo di installazione vera e propria, sarà possibile eseguire ulteriori verifiche sui componenti. Per esempio, si può controllare l'integrità del CDROM, salvare i registri di debug o eseguire una shell.

Non manca niente

L'interfaccia grafica che vi accoglie all'avvio di Tango Studio è particolarmente minimale. I toni scuri consentono di interagire perfettamente con l'ambiente senza distrazioni. La barra degli strumenti è collocata nella parte superiore dello schermo, da cui si può accedere a tutte le funzioni più importanti. Le tre voci presenti offrono scorciatoie rapide alle applicazioni, alla gestione del disco e alle funzioni utili alla configurazione del sistema. Tra i programmi presenti, oltre a quelli dedicati all'audio e video di cui parleremo tra poco, troviamo LibreOffice, Deluge, Icedove Mail/News, Iceweasel, Internet DJ Console, Pidgin, GIMP, gThumb, Eye of Mate e altri. Ci sono anche una serie di strumenti di sistema, come UXTerm, XTerm e Htop. Tango Studio supporta poi la connessione remota. Basta scegliere la voce **Connetti al server** e impostare il servizio a cui volete collegarvi: FTP, SSH, Windows share, WebDAV o Secure WebDAV. Non manca Wine per utilizzare eventuali applicazioni Windows dedicate al comparto audio e video. La sezione multimediale è comprensibilmente quella più nutrita. La troviamo suddivisa in varie sezioni, tra cui quella dedicata ai MIDI, alla scrittura e alla riproduzione dei CD/DVD, alla gestione degli effetti musicali, alla registrazione, alla gestione della scheda audio e all'uso di vari sintetizzatori e sampler. Grazie alla sezione JACK, possiamo poi gestire qualsiasi strumento si voglia collegare al computer. Con Tango Studio, infatti, siamo in grado di realizzare a costo zero un sistema base per uno studio musicale. Vista nel suo complesso, è comunque una distribuzione particolarmente completa, che permette di essere utilizzata anche durante la normale attività quotidianità.

Cosa c'è nel DVD

DISTRIBUZIONI
- GhostBSD 4.0
- Tango Studio 2.2 i386
- Tango Studio 2.2 x64
- KXStudio 14.04

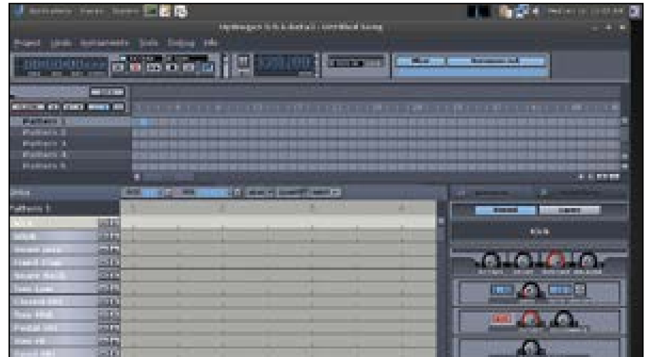
PROGRAMMI
- Dr. Geo 14.12
- Fame 0.96
- Guacamole 0.9.3 client
- Guacamole 0.9.3 server
- i3 4.8
- QMMP 0.8.3
- Stumpwm 0.9.8
- TMux 1.9a
- Untangle

Caratteristiche Cosa ci offre Tango Studio 2.2



1 Tutto a portata di mano

Il desktop di Tango Studio 2.2 non lascia spazio a distrazioni. Mette subito al centro dell'attenzione ciò che serve. In questo caso vediamo il menu **Sound & Video** che è anche quello più completo e ricco di applicazioni. Dalla barra degli strumenti, sono comunque accessibili altre funzioni relative alla gestione del sistema e dei dischi fissi.



2 Drum Machine

Tra i vari programmi che troviamo in Tango Studio c'è anche **Hydrogen**. Si tratta di una drum machine estremamente avanzata, spesso utilizzata anche dai professionisti dell'audio. Consente di creare composizioni estremamente elaborate e in più di modificare quelle esistenti. Per usarla ci vuole comunque esperienza.



3 Sequencer MIDI

musE è un'altra applicazione che troviamo nella sezione dedicata ai MIDI. Si tratta di un sequencer audio basato su Qt, che consente di modificare e registrare una grande quantità di suoni. Se non sapete da dove iniziare a usarlo, un buon punto di partenza in cui trovare documentazione è il sito ufficiale **http://muse-sequencer.org**.



4 Dedicato ai chitarristi

Se siete degli appassionati di chitarra elettrica, apprezzerete la presenza di **Guitarix**, un amplificatore virtuale che lavora con JACK. Tra le sue peculiarità più interessanti c'è la completa compatibilità con qualsiasi altra applicazione *Jack Audio Connection Kit*. Questo ne fa uno strumento estremamente versatile da usare.



5 Musica maestro!

MuseScore è un'altra applicazione estremamente potente. Dedicata alla notazione e alla partitura dei brani, consente ai compositori di sbizzarrirsi nei più svariati modi. Anche in questo caso non si tratta di un programma semplice da usare, soprattutto se siete dei principianti. Per le basi, potete dare un'occhiata a **http://musescore.org/it**.



6 Sequencer in Qt

Tra i vari sintetizzatori e sequencer disponibili abbiamo trovato **Qsynth**. Si tratta di un'interfaccia Qt per FluidSynt. Scritto in C++ su Qt4, permette di sintetizzare in tempo reale qualsiasi fonte sonora. È particolarmente potente e versatile da usare. Se volete maggior informazioni sul programma, collegatevi a **http://qsynth.sourceforge.net**. LXP

In edicola il 12 marzo

NEL PROSSIMO NUMERO

Hai un argomento da proporci?
Scrivi a prossimamente@linux.it

CREA IL MEDIA SERVER PERFETTO

Ecco tutti i consigli per scegliere e configurare la distribuzione perfetta per gestire musica, film, serie TV e foto da qualsiasi dispositivo!



Mensile, prezzo di copertina 5,90 €
www.linuxpro.it

Direttore responsabile: *Luca Sprea*

Localizzazione cura di:
Ventidodici di Andrea Orchesi
redazione@linuxpro.it


Sprea S.p.A.
Socio Unico - direzione e coordinamento di Sprea Holding S.p.A.

Presidente: Luca Sprea
Consigliere delegato: Mario Sprea

Coordinamento:
Gabriella Re (Foreign Rights) international@sprea.it, Alberta Rivolta (PA), Ambra Palermi (Segreteria Editoriale), Francesca Sigismondi (Ufficio Legale), Tiziana Rosato (acquisti e produzione), Emanuela Mapelli (Pianificazione Pubblicitaria)

Amministrazione: Anna Nese (CFO), Erika Colombo (controller), Irene Citino, Sara Palestra
amministrazione@sprea.it

Servizio qualità edicolanti e DL: Sonia Lancellotti, Andrea Palermi
distribuzione@sprea.it

PUBBLICITA' E MARKETING:
Alessandra Cappellacci- alessandracappellacci@sprea.it
Tel.0292432275 - Cell.3204670523

Sede Legale: - via Torino, 51 20063 Cernusco Sul Naviglio (Mi) - Italia
PI 12770820152- Iscrizione camera Commercio 00746350149

ABBONAMENTI E ARRETRATI

Servizio qualità abbonamenti e arretrati: Desirée Conti

Abbonamenti: si sottoscrivono on-line su www.linuxpro.it/abbonamenti
Mail: abbonamenti@linuxpro.it
Fax: 02 700 53 76 72
Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)

Arretrati: si acquistano on-line su www.linuxpro.it/arretrati
Mail: arretrati@linuxpro.it
Fax: 02 700 53 76 72
Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)

www.myabb.it a cura di Aktia srl

Registrazione testata: Linux Pro, pubblicazione mensile registrata al Tribunale di Milano il 08.02.2003 con il numero 74.

Distributore per l'Italia e per l'estero:
Press-Di Distribuzione stampa e multimedia s.r.l. - 20134 Milano
ISSN: 1722-6163

Stampa: Arti Grafiche Boccia S.p.A.- Salerno

Copyright Sprea S.p.A.
La Sprea S..p.a è titolare esclusiva della testata Linux Proe di tutti i diritti di pubblicazione e di diffusione in Italia. L'utilizzo da parte di terzi di testi, fotografie e disegni, anche parziale, è vietato. L'Editore si dichiara pienamente disponibile a valutare - e se del caso regolare - le eventuali spettanze di terzi per la pubblicazione di immagini di cui non sia stato eventualmente possibile reperire la fonte. Informativa e Consenso in materia di trattamento dei dati personali (Codice Privacy d.lgs. 196/03). Nel vigore del D.Lgs 196/03 il Titolare del trattamento dei dati personali, ex art. 28 D.Lgs. 196/03, è Sprea S.p.A. (di seguito anche "Sprea"), con sede legale inVia Torino, 51 Cernusco sul Naviglio (MI). La stessa La informa che i Suoi dati, eventualmente da Lei trasmessi alla Sprea, verranno raccolti, trattati e conservati nel rispetto del decreto legislativo ora enunciato anche per attività connesse all'azienda. La avvisiamo, inoltre, che i Suoi dati potranno essere comunicati e/o trattati (sempre nel rispetto della legge), anche all'estero, da società e/o persone che prestano servizi in favore della Sprea. In ogni momento Lei potrà chiedere la modifica, la correzione e/o la cancellazione dei Suoi dati ovvero esercitare tutti i diritti previsti dagli artt. 7 e ss. del D.Lgs. 196/03 mediante comunicazione scritta alla Sprea e/o direttamente al personale Incaricato preposto al trattamento dei dati. La lettura della presente informativa deve intendersi quale presa visione dell'Informativa ex art. 13 D.Lgs. 196/03 e l'invio dei Suoi dati personali alla Sprea varrà quale consenso espresso al trattamento dei dati personali secondo quanto sopra specificato. L'invio di materiale (testi, fotografie, disegni, etc.) alla Sprea S.p.A. deve intendersi quale espressa autorizzazione alla loro libera utilizzazione da parte di Sprea S.p.A. Per qualsiasi fine e a titolo gratuito, e comunque, a titolo di esempio, alla pubblicazione gratuita su qualsiasi supporto cartaceo e non, su qualsiasi pubblicazione (anche non della Sprea S.p.A.), in qualsiasi canale di vendita e Paese del mondo.
Il materiale inviato alla redazione non potrà essere restituito.

“Non condivido le tue idee, ma sono pronto a morire affinché tu le possa esprimere”
Voltaire

JE SUIS
CHARLIE

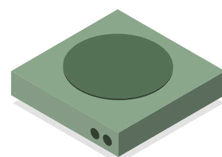
“L’opinione è forse il solo cemento della società”
Cesare Beccaria

L'Hosting a km 0

Data Center
2 IN ITALIA
6 in Europa

Per la tua attività online scegli la garanzia del Made in Italy.

L'Hosting di Aruba è affidabile, potente e completamente personalizzabile.
Puoi scegliere tra moltissimi servizi opzionali, tra cui database, statistiche e backup, per creare la soluzione hosting su misura per il tuo progetto. E grazie all'ottima connettività Aruba, il tuo sito è veloce sia dall'Italia che dall'estero.



Spazio disco e traffico illimitato



1 dominio incluso
con estensione a tua scelta



5 caselle email
da 1GB incluse



Possibilità di e-commerce

A partire da
20,66 € + IVA/anno

In più, con Application Installer installare CMS e app è facile e veloce.









Per maggiori informazioni: www.aruba.it 0575 0505 *assistenza in italiano*

aruba.it

Hosting Linux e Windows | Hosting Managed | Hosting Personalizzato | Hosting Plesk | Hosting cPanel